About this database

Sms4Domino is an application that delivers SMS (short messaging service) capabilities to Domino applications. End users send and receive SMS messages the same way as they send and receive e-mail messages.

The application is fully integrated in the Domino environment. Domino administrators will feel at ease when configuring and monitoring the application.

The application performs all processing in Lotus Domino® databases. There is no need for additional database servers or specialised skill sets.

Sms4Domino consists of one or more Lotus Domino® databases. In small environments the databases may be combined into one single database.

- Configuration database (.nsf) that defines the gateways to the GSM network, the interfaces that generate SMS requests, the license, the configuration of the logging, and all required parameters.

- Accounting database (.nsf) where historical data is stored. Accounting records may be used to restrict people from sending SMS messages.

- SMS mailbox databases. The Domino router stores SMS requests into this database. Sms4Domino polls the database, converts the requests into SMS messages and transmits them via one of the defined gateways.

- Sms4Domino run time program. This Java program may run as a Domino server add-in on a Windows based Domino server or as a standalone program on a workstation on which Lotus Notes® has been installed.

Sms4Domino is build as a Java program on top of different well known open source packages:

- smslib: Java toolkit for sending and receiving Sms messages (http://smslib.org)

- log4j: Java logging framework (http://logging.apache.org/log4j/1.2/)

- JAddin: Java toolkit for Domino Add-In Server tasks (http://abdata.ch/JAddIn.html)

- libphonenumber: Google's phone number handling library, powering Android and more (http://code.google.com/p/libphonenumber)

The open source packages are distributed under the terms of the Apache V2 license. The code and the licensing terms are stored in the Sms4Domino template.

SMS messages are sent via one or more GSM modems connected to the server or workstations. We strongly recommend using industrial grade modems connected via a serial COM port. (Only the Cinterion MC55iT modem is supported).

Sms4Domino has been developed on a Domino 8.5 platform on Windows. The program depends on the Java run time environment supplied with Domino or Notes. This must be at least version ....

Please visit us at http://www.sms4domino.eu for more information. All your feedback is welcome.

Revision history

The version of the software is printed at the console when the application is started.

Version	Description
3.3 (January 2013)	- New features ● Textual error code for Cinterion modem ● Possibility to reply from the user's mailbox to received SMS ● More granular configuration of inbound routing
3.0 (September 2012)	- New features ● Multi-server implementations ● Clustered implementations ● Least cost routing ● Creation of accounting records for inbound and outbound traffic
2.0 (May 2012)	- New features ● Support for Clickatell and BulkSms gateways (send only) ● Evaluation license ● Possibility to use GSM phone (for evaluation/testing purpose)
1.0 (March 2012)	- Initial production release ● Inbound and outbound SMS messages ● Compatible with Lotus Traveler® for implementing SMS push ● Inbound routing via fixed routing or auto matching ● Restrictions on allowed destination phone numbers ● Content of outbound SMS message may be changed at run-time via @formula ● For use with Cinterion (Siemens)® MC55iT GSM Quad Band Modem
0.1 (January 2012)	- Initial internal release ● Only outbound SMS messages

System architecture

For small configurations there will be one instance of the Sms 4Domino program.

In order to handle heavier workloads, to perform least cost routing, to increase service availability it may be required to roll-out more than one Sms4Domino instance on multiple Domino servers.

First the single instance configuration is described. This will then be used as a building block for creating a multi-instance set-up.
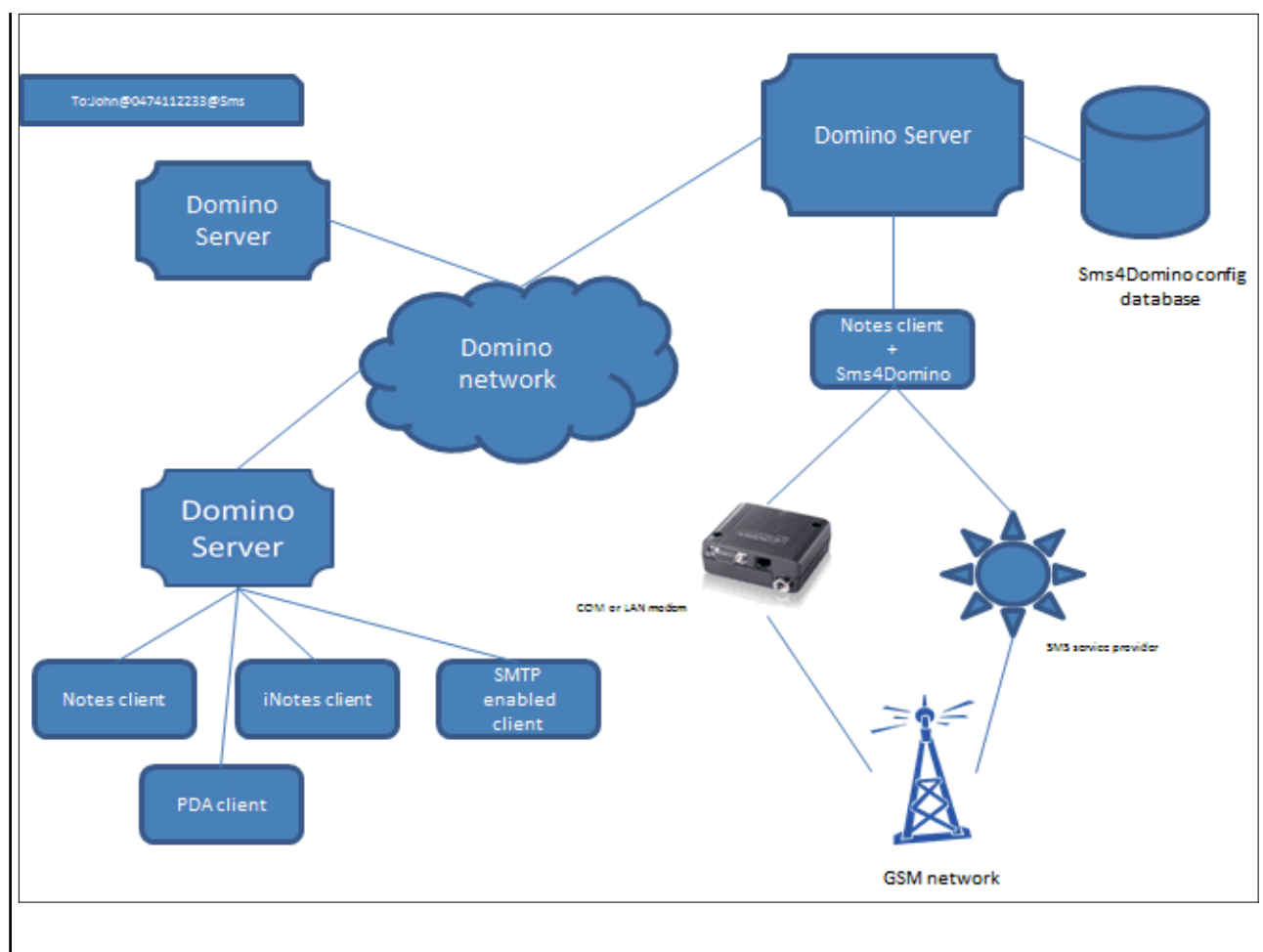
Single Instance system architecture

Sms4Domino integrates in any Domino infrastructure. Following drawing illustrates the architecture when Sms4Domino is running as a server addin. Typically the configuration database and the Sms4Domino program are located on the same server. One or more modems are connected via either COM ports or via the LAN to the Domino server. Sms4Domino may also connect to a SMS service provider (e.g. Clickatell) over HTTP.

End users and systems may send mail from different kind of workstations anywhere in the network. Domino will route the mail to Sms4Domino.

The running Sms4Domino program will be called a Sms4Domino instance. In this diagram the instance is running with the credentials of the Domino server.
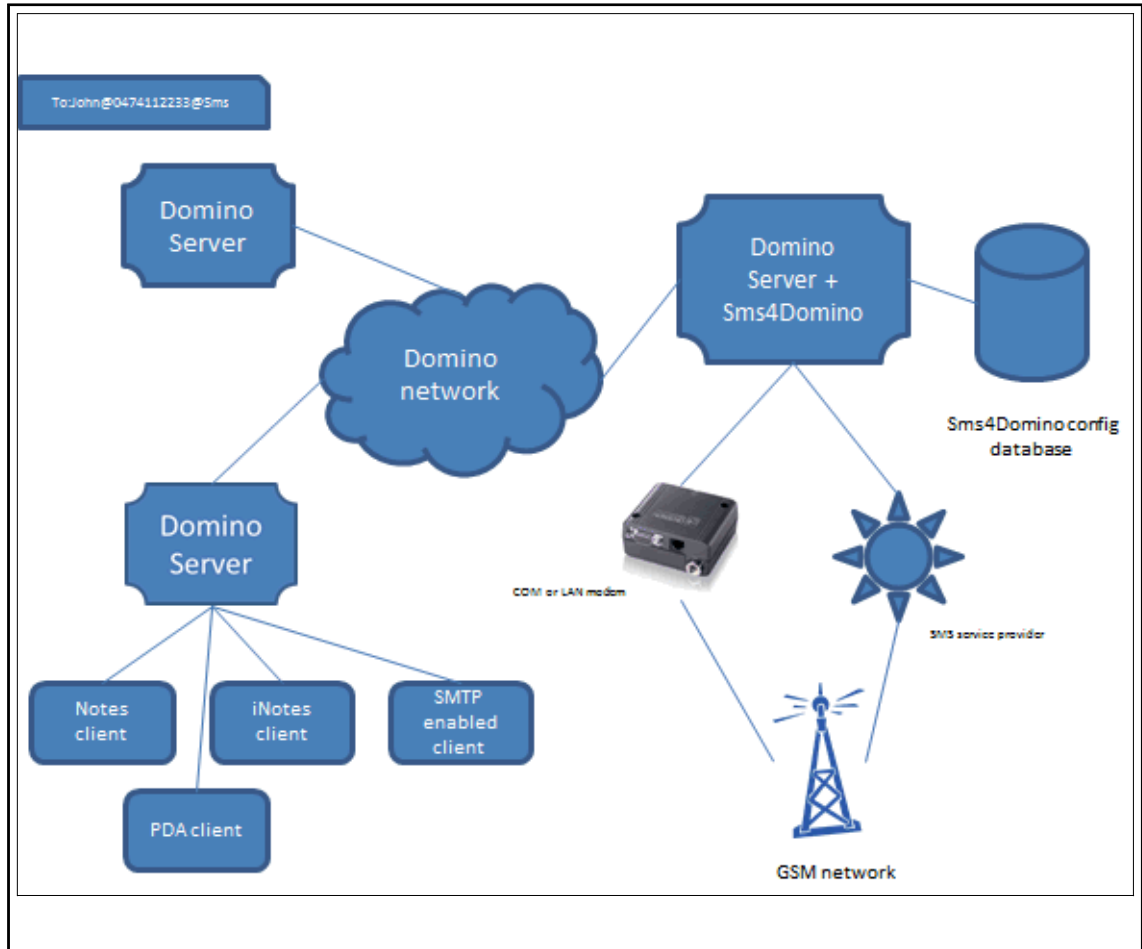
As an alternative Sms4Domino may run on a Windows workstation where the Lotus Notes® 8.x client has been installed. The modems are connected to the COM ports of the workstation or to the LAN. The configuration database resides on any of the Domino® servers.
In this diagram there is also only one Sms4Domino instance running with the credentials of the user.id defined by the notes.ini file of the workstation.

Multi Instance system architecture

Why a multi instance architecture

The main reasons for working in a multi-instance configuration are

- Improved reliability: when one Domino server becomes unavailable users may continue sending SMS messages via one the other servers. Domino routing and replication may be configured in such a way that this is transparant for them.

- Least cost routing: in a multi-country environment it may be cheaper to use Domino replication to route the request to a server that has a low cost Sms connection to the requested destination. The configuration of the least-cost routing is very flexible and takes into account the phone cost, the replication cost and the workload of the instances.

- Clustering: mail is considered as a critical service by many companies. Domino clustering is frequently used in these environments. Sms4Domino may also be used in a clustered environment offering improved reliability and throughput.

Building blocks

In the multi-instance configuration the Sms4Domino databases are replicated over different Domino servers. By preference cluster based replication is used. Every replica of the configuration database has a corresponding Sms4Domino instance. This instance runs either on the Domino server (server add-in) or on a Notes client (standalone version). Each Sms4Domino instance manages zero, one or more GSM modems.

Example of processing a SMS request

Company ACME has offices in France, Germany and the UK. There is a Domino server in each of the countries. To benefit of least cost SMS routing Sms4Domino has been installed on each server.

User John, working in the UK, would like to send a Sms to his frient Paul. Paul lives in France. John sends a mail to Paul@+33477858589@sms. Domino mail routing delivers the mail in the Sms4Domino mailbox on John's home server. Sms4Domino picks up the request, analyzes the phone number and calculates the lowest cost route for sending the SMS. The server in France should do the actual sending. Sms4Domino-UK stores the best route in the Notes document. The document replicates to the server in France (this replication happens within a few seconds in case of a Domino cluster; if clustering is not available Sms4Domino-UK will initiate the replication to Sms4Domino-FR).

Sms4Domino-FR sees the new request and sends the SMS via the local GSM modem.

Limitations of this release

Some features that are described in the help file or on the configuration settings have not yet been implemented.

- Sms4Domino may be rolled-out as a multi-database application (configuration database, accounting database, ...). In the current release all Sms4Domino features must be part of a single configuration database. This database may be replicated to multiple servers.

- Not all Interface families have been implemented. In the current release requests for sending a SMS must be mailed to Sms4Domino.

- Only one Notes Mail Interface may be defined. This implies that there is only one foreign domain document related to SMS messages.

- Not all gateway families that are supported by SmsLib are available for being used by Sms4Domino. Currently we recommend connecting an industrial modem via a serial cable to a COM port. Multiple modems may be connected to the same machine. All modems must be connected to the machine where Sms4Domino is running.

- As an alternative to SMS modems it is possible to make use of SMS service providers (Clickatell or BulkSms). These gateways are outbound only.

30 minute installation guide

The installation should be straightforward for a Lotus Domino® system administrator. We will give overviews for deploying a single instance or multi instance configuration.

Before you begin

- Acquire one or more industrial grade GSM modems (e.g. Cinterion® MC55iT) and corresponding SIM cards. (During the development of Sms4Domino we tested many different modems. Although they should all implement the same set of AT commands we have experienced subtle differences in the way this is done. Sms4Domino support for modem related issues is therefore available for this modem only.) For evaluation purposes you could consider using a mobile phone and connect it to the USB port of the computer.

- Connect the modem(s) to the serial port(s) on the computer. Consider installing a PC card with multiple serial ports. (It is also possible to connect the modem, via a serial-to-ethernet convertor, directly to the LAN. This setup is documented in a separate help document)

- Test the communications between the PC and the SMS modems. Windows provides some basic test tools.
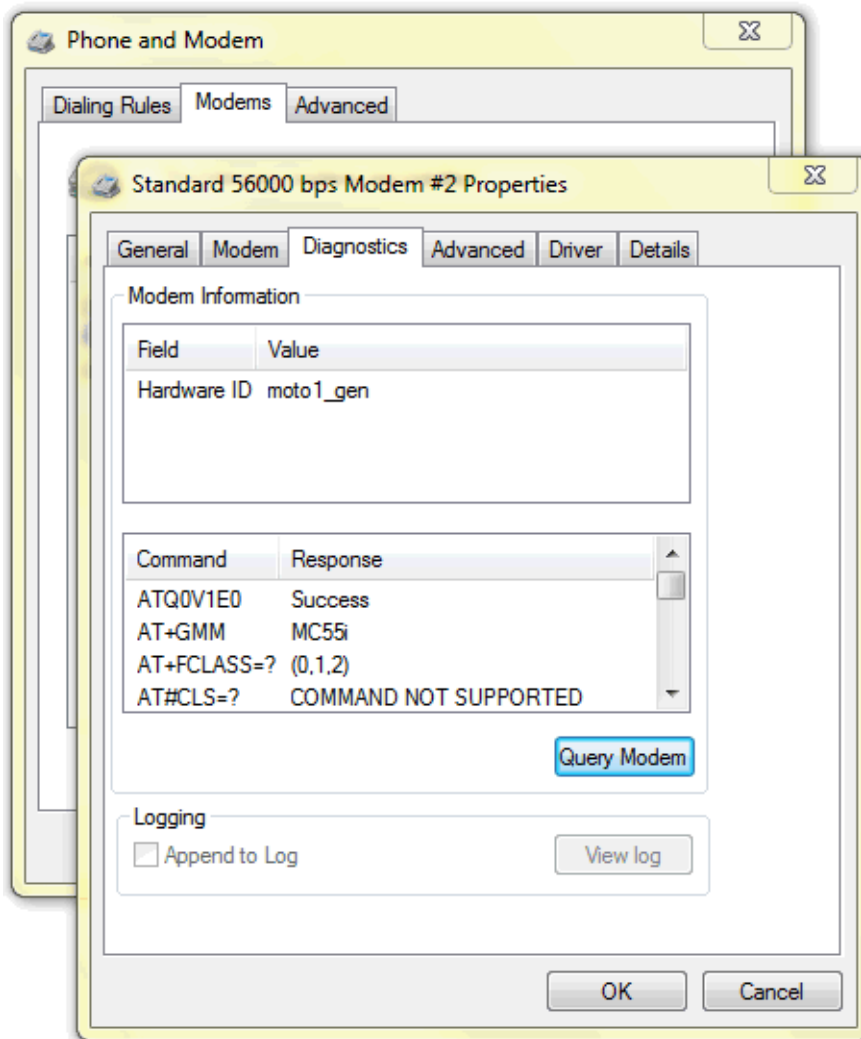
- Prepare the modem for use with Sms4Domino.

- Decide on the architecture. Sms4Domino may run as a Domino server add-in (started via

RunJava) or as a standalone program on a PC where Lotus Notes® clients has been installed.

Installation steps

This document contains only a checklist of all installation steps . More information is available in subsequent documents.

1. Obtain a copy of the Sms4Domino template. Sign the template with a proper user id. Adapt the ACL to your local standards.

2. Create a configuration database on a Domino server. (The database must always reside on a server - not on a workstation). Adjust the ACL settings.

    *   For multi instance configurations the database must be replicatedto all participatings Domino servers.

3. Extract the dll and jar files from the configuration database and install them in the appropriate directories on the server or workstation where Sms4Domino will be running.

4. Add some mandatory parameters to the notes.ini of the server or workstation where Sms4Domino will be running.

5. Define the basic parameters in the application profile of the configuration database .

6. Create and configure one or more Sms4Domino instance documents.

7. Create and configure gateway documents. Every transmission path to the GSM network requires a gateway document.

8. Create and configure interface documents. Each source of SMS requests requires an interface document. Domino mail interfaces may receive the Domino mails in the configuration database or in a separate database. In all cases it is the most practical to base the databases on the Sms4Domino template.

9. Optional: enable the pre-mail delivery agent *premaildelivery*

10. Create and configure one or more routing rules documents. Even for a single instance configuration the should be a routing rule document.

11. Create and configure a license document for every Instance. Obtain a valid license file from Sms4Domino and copy the contents of the file into the license document.

12. Create and configure logging documents. Logging is based on the Apache log4J software. Logging may be very tearse of verbose. Output of the logging may be directed to the Domino log.nsf or to a Windows file.

13. Define a foreign domain in the Domino directory . Notes users will address mails to John@3618319@sms or to 3618319@sms for sending a SMS message to this number. The mailbox of the foreign domain points to the Sms4Domino control database or another database that has been defined in the interface document.

14. Start the Sms4Domino program.

The system is now ready to be used. The administrator should

*   Monitor the system.
*   Perform maintenance actions.

Choice and installation of the modem or Sms service provider

Sms4Domino is running on top of SmsLib. This software is responsible for the communication with the modem. SmsLib supports different kind and models of modems like a USB connected GSM, an industrial modem connected via a serial COM port, a modem directly connected to the LAN or a SMS service provider like Clickatel.

In the current release Sms4Domino supports a local or LAN connected modem or a connection via a service provider. We recommend using an industrial grade modem connected to a serial port (not via USB, bluetooth or infrared). During testing we have noticed that not all modems are equal. They may behave in unexpected ways under certain conditions. In order to receive support for modem related questions you should use the Cinterion MC55iT modem (http://www.cinterion.com). In case the modem must be connected to the LAN we recommend the use of the ABLELink Ethernet Serial Server (Atop technologies - http://www.atop.com.tw)

Modern PCs have no are only one COM port. It may be necessary to install an expansion board providing one, two or more additional COM ports. These cards come mostly with driver software. After a successfull installation the COM ports must be visible in the Windows device manager
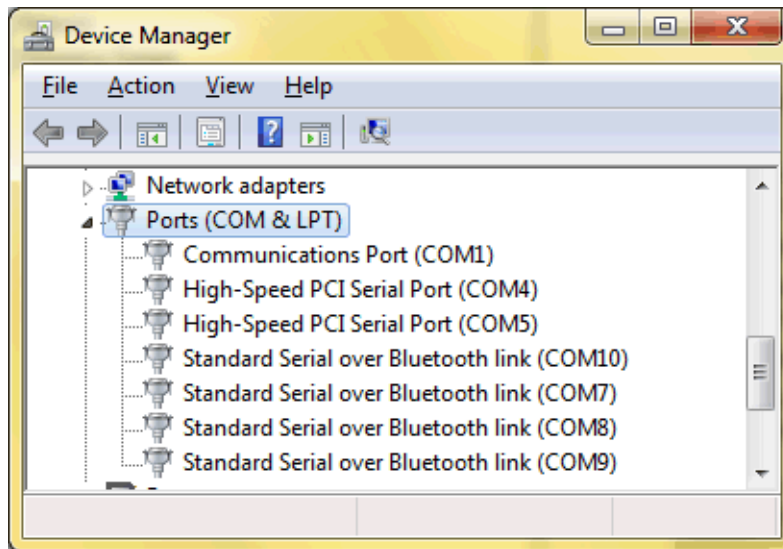
After this verification it is possible to connect the modem to the COM port. Read the documentation supplied with the modem. You must know the communication speed and parameters between the PC and the modem. Sms4Domino will communicate with the modem at a configurable speed. However Sms4Domino will communicate with following settings

- Speed: defined in the gateway document. Modems operate either at a fixed bit rate or may recognize automatically the bit rate being used. This auto bit rate recognition feature does not work if the speed is too high or too low. Consult the documentation supplied with the modem.
- 8 databits
- 1 stop bit
- no parity bit
- Flow control based on RTS and CTS

When the modem is connected to the PC it is recommended to perform some basic communication tests. There are different ways for doing this. We explain a few of them.
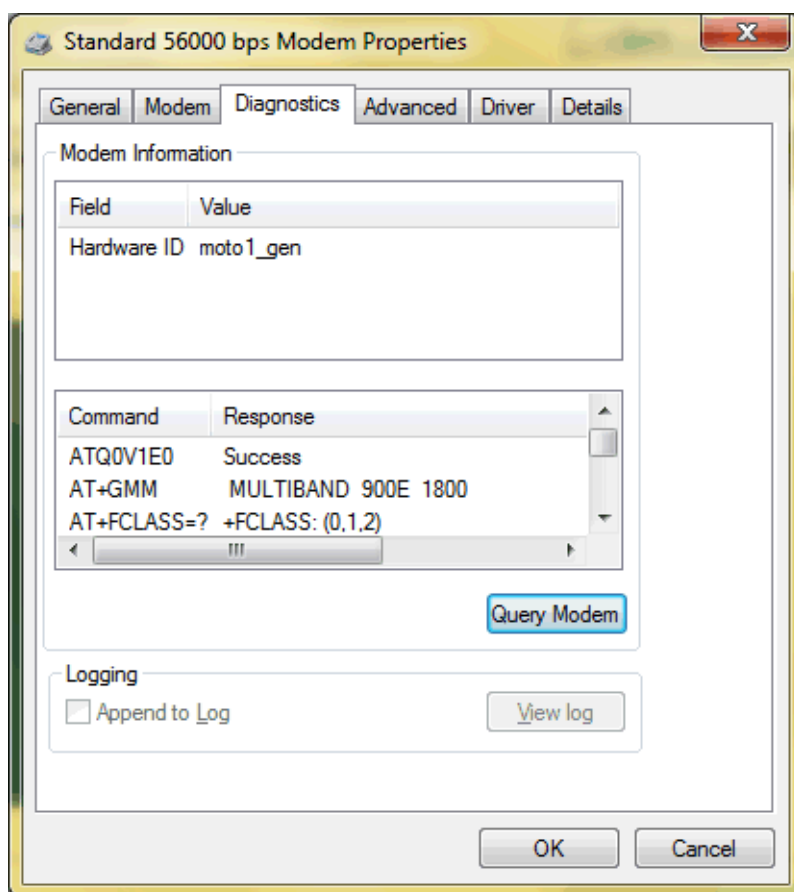
Windows modem test

Open the Windows control panel. Click on *Phone and modems.* Add a generic modem to the COM port (COM5 in the print screen)

Phone and Modem

Dialing Rules | **Modems** | **Advanced**

The following modems are installed:

Modem	Attached To
Standard Modem over Bluetooth link	COM3

Add a modem

Add... | Remove | Properties

OK | Cancel | Apply

Add Hardware Wizard

Install New Modem
Do you want Windows to detect your modem?

Windows will now try to detect your modem. Before continuing, you should:

1. If the modem is attached to your computer, make sure it is turned on.

2. Quit any programs that may be using the modem.

Click Next when you are ready to continue.

☑ Don't detect my modem; I will select it from a list.

< Back | Next > | Cancel

Add Hardware Wizard

Install New Modem

Select the manufacturer and model of your modem. If your modem is not listed, or if you have an installation disk, click Have Disk.

Manufacturer	Models
(Standard Modem Types)	Standard 28800 bps Modem
Atheros	Standard 33600 bps Modem
	Standard 56000 bps Modem
	Standard PCMCIA Card Modem

This driver is digitally signed.

Tell me why driver signing is important

Have Disk...

< Back   Next >   Cancel

Add Hardware Wizard

Install New Modem

Select the port(s) you want to install the modem on.

You have selected the following modem:

Standard 56000 bps Modem

On which ports do you want to install it?

○ All ports
⊙ Selected ports

COM1
COM4
COM5
COM8
COM10

< Back   Next >   Cancel

The modem may be tested via the query tool under the Diagnostics tab .

COM test program

The next example is based on the Advanced Serial Port monitor (evaluation version available on http://www.aggsoft.com).
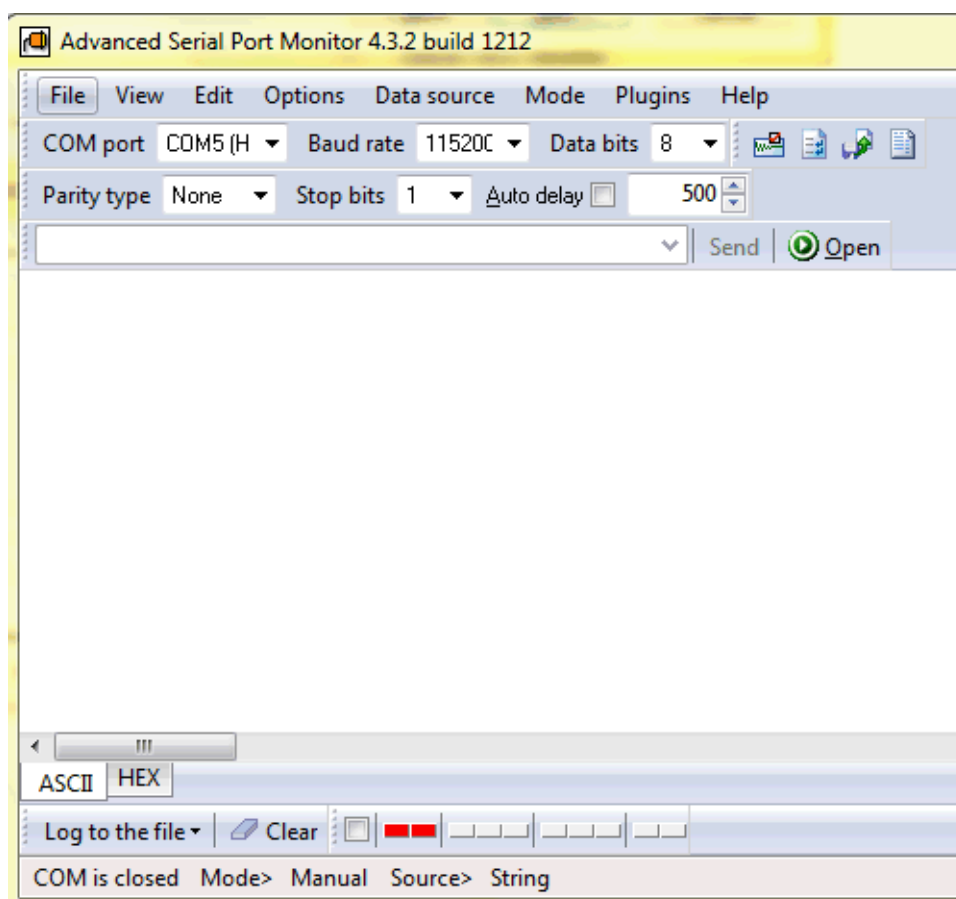
The program must be started in administrator mode. Open the port at the speed configured in the modem.

Send the AT command to the modem. The modem should respond with OK.

Putty (http://www.putty.org)
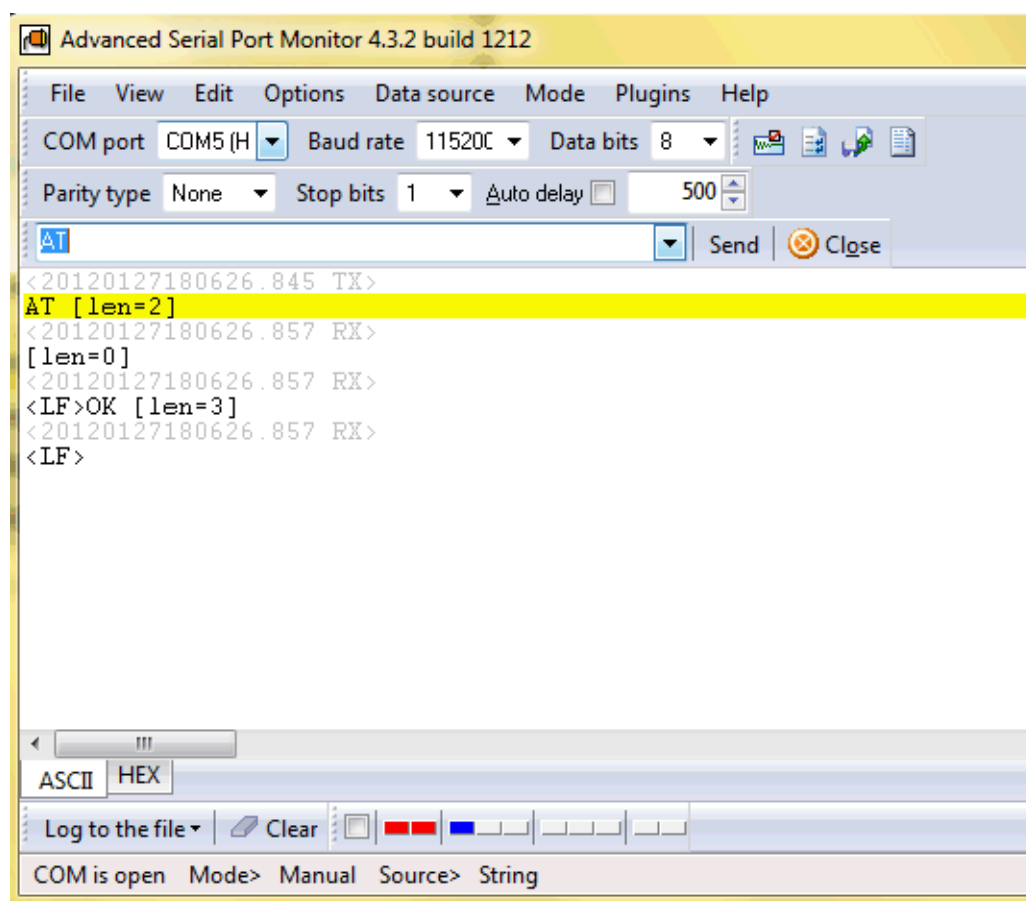
Putty is a free open source telnet client. It is normally used to communicate with devices connected to the LAN. However, in serial mode, it is perfectly suited to test the modem connected to the com port.

Download putty to your local Windows machine. Executy the putty.exe program
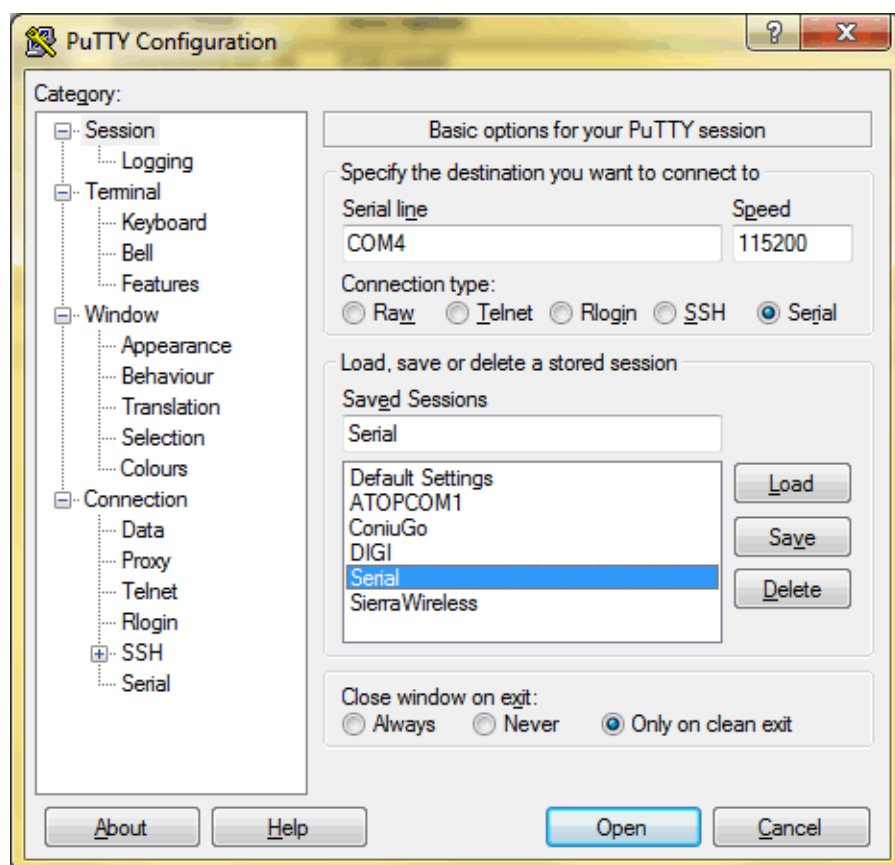
Configure your session (Clickon Session in the left pane)

     Connection type = serial
     Serial line = the COM port you are using
     Speed = the default speed for your modem
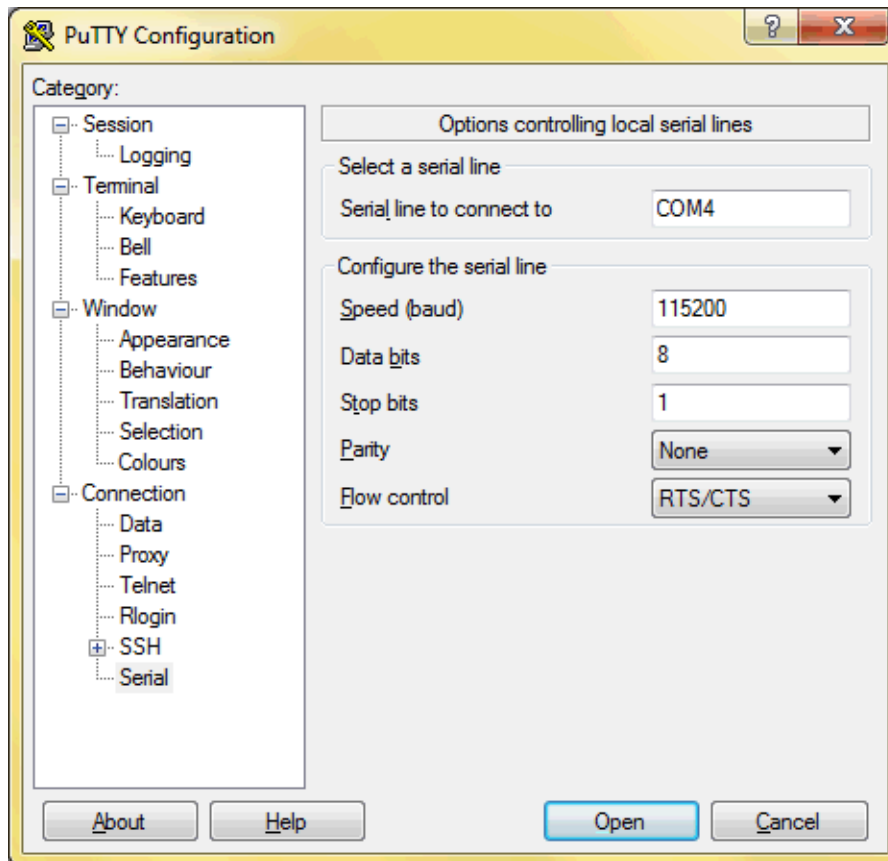
Specify additional settings for the serial connection (click on Connection -> Serial in the left pane)

Data bits: 8
Stop bits: 1
Parity: None
Flow control: RTS/CTS

You are now ready to test the connection with your modem. Click on open. The modem should respond with OK to the AT command

Initial configuration of the Cinterion MC 55iT modem

Sms4Domino assumes the modem has been properly initialized. This is true for a Cinterion modem that uses the factory default settings. However, when a modem is moved from some other project to Sms4Domino it may be necessary to change some modem settings.

Changing the settings is done by sending AT commands to the modem. We prefer connecting the modem directly to a serial port. Trying to connect the unknown modem to the serial-to-Lan convertor and then trying to talk to the modem over the LAN is not recommended.

This document assumes you are using Putty for communicating with the modem.

Following steps will be executed

- Establish communication with the modem. It may be necessary to guess the baud rate
- Reset the modem to the factory defaults
- Set the required modem parameters
- Save the parameters to non volatile RAM. After each reboot of the modem these settings will be loaded automatically into memory.
- Verify the PIN code of the SIM card

Establish communication with the modem

Find in the office a computer with a COM (RS-232) port. Determine the identifier of the COM port. In most cases this will be COM1. The Windows device manager lists all known ports
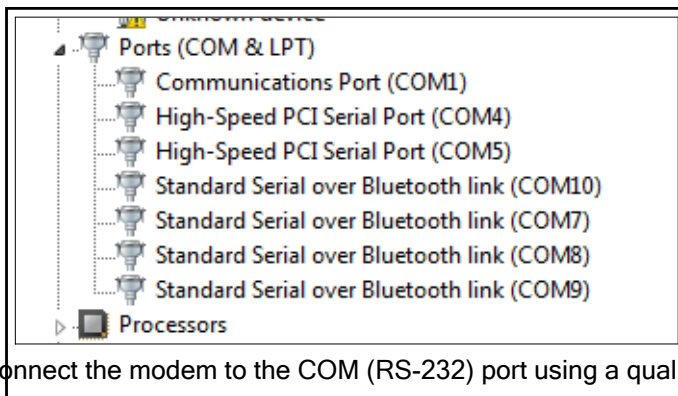
Connect the modem to the COM (RS-232) port using a quality RS-232 cable.

Launch the putty program
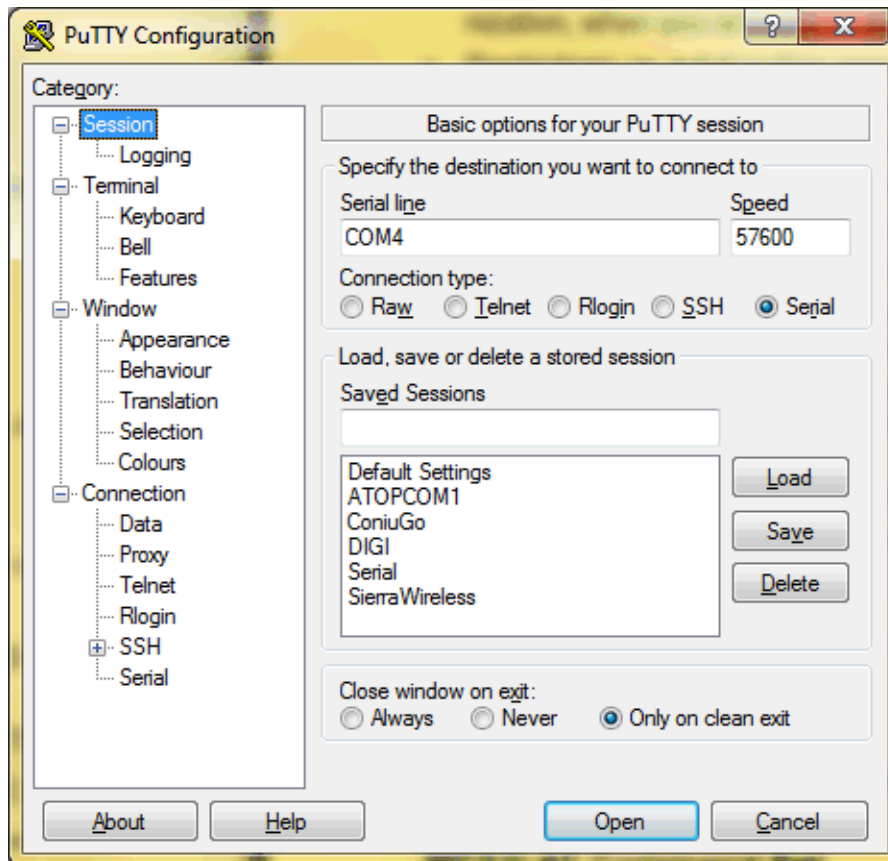
Within the Session category select the COM port (COM4 in above example) and the speed.
57600 bps will be recognized by the modem if AutoBauding is still enabled. Acceptable speeds
for the Cinterion modem are 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 28800, 38400,
57600, 115200 and 230400 bits per second. AutoBauding is only possible for speeds up to
115200 bps.
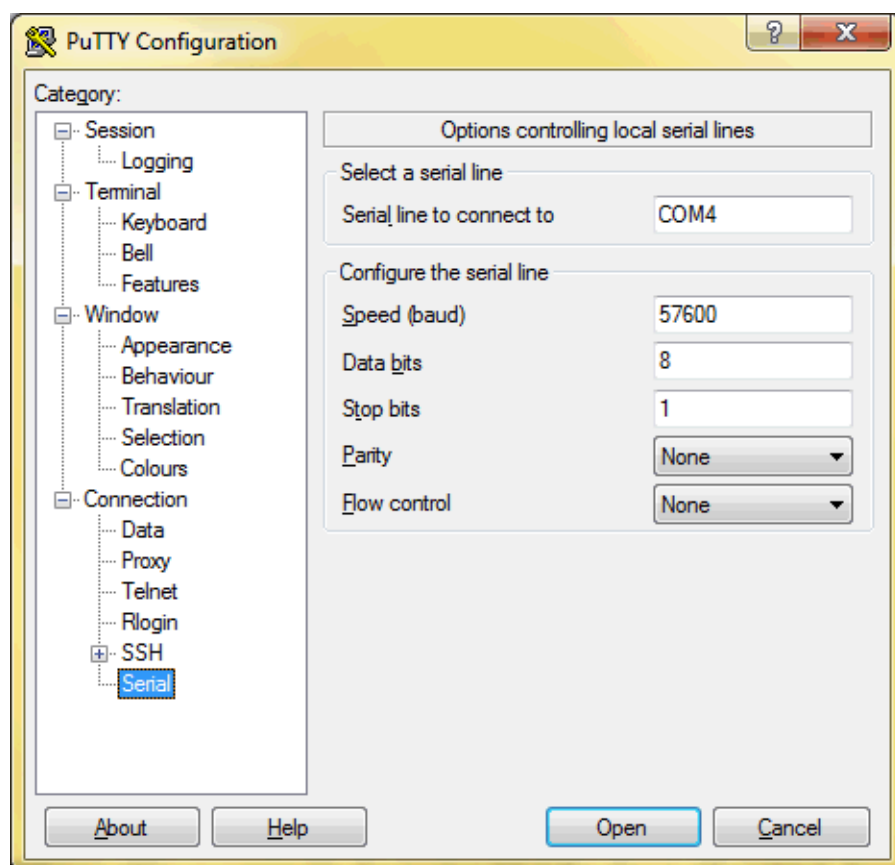
Within the Connection -> Serial category specify the settings for data bits, stop bits, parity and flow control. Factory default settings are 8 data bits, 1 stop bit, no parity and no flow control.

Click Open. A pop-up window is displayed.

Enter the letters AT followed by pressing *Return.* The could be an echo of AT. The modem should respond with OK
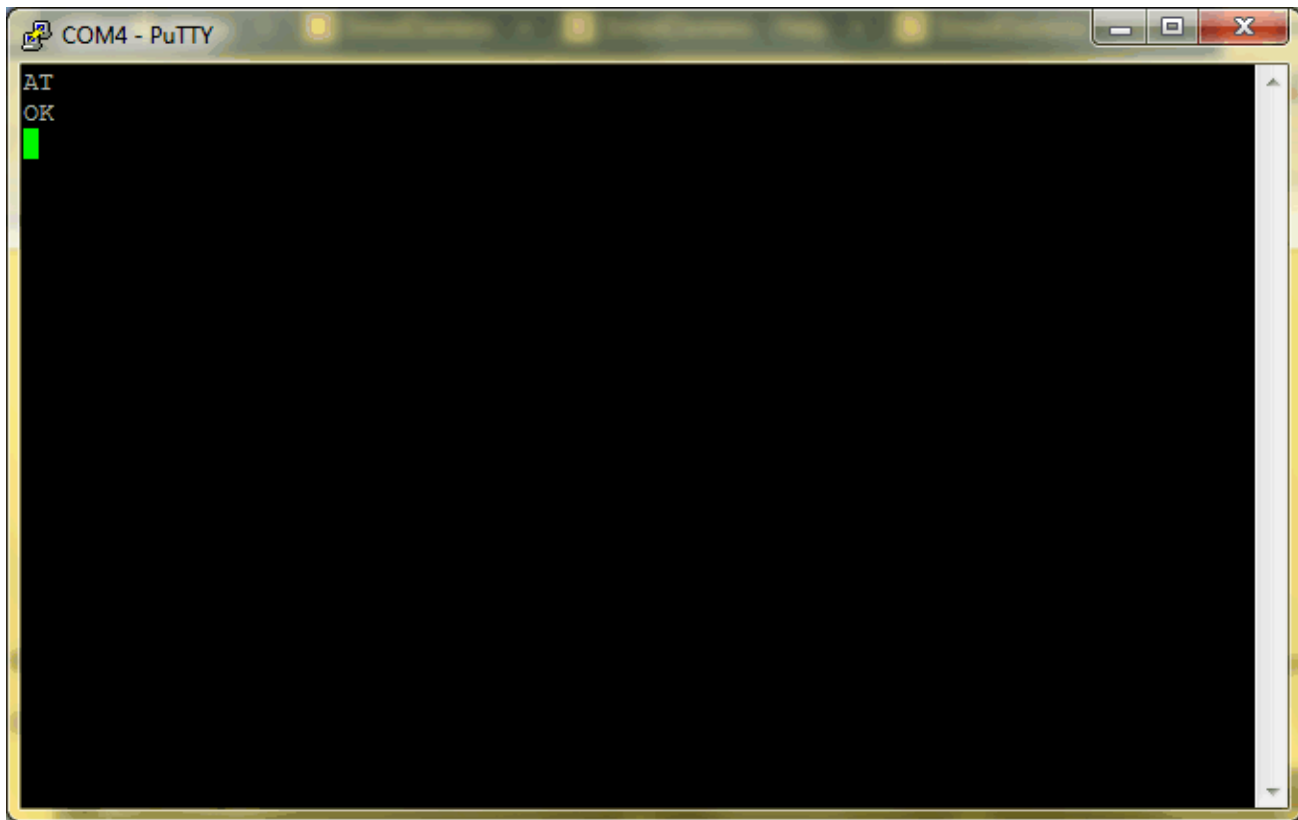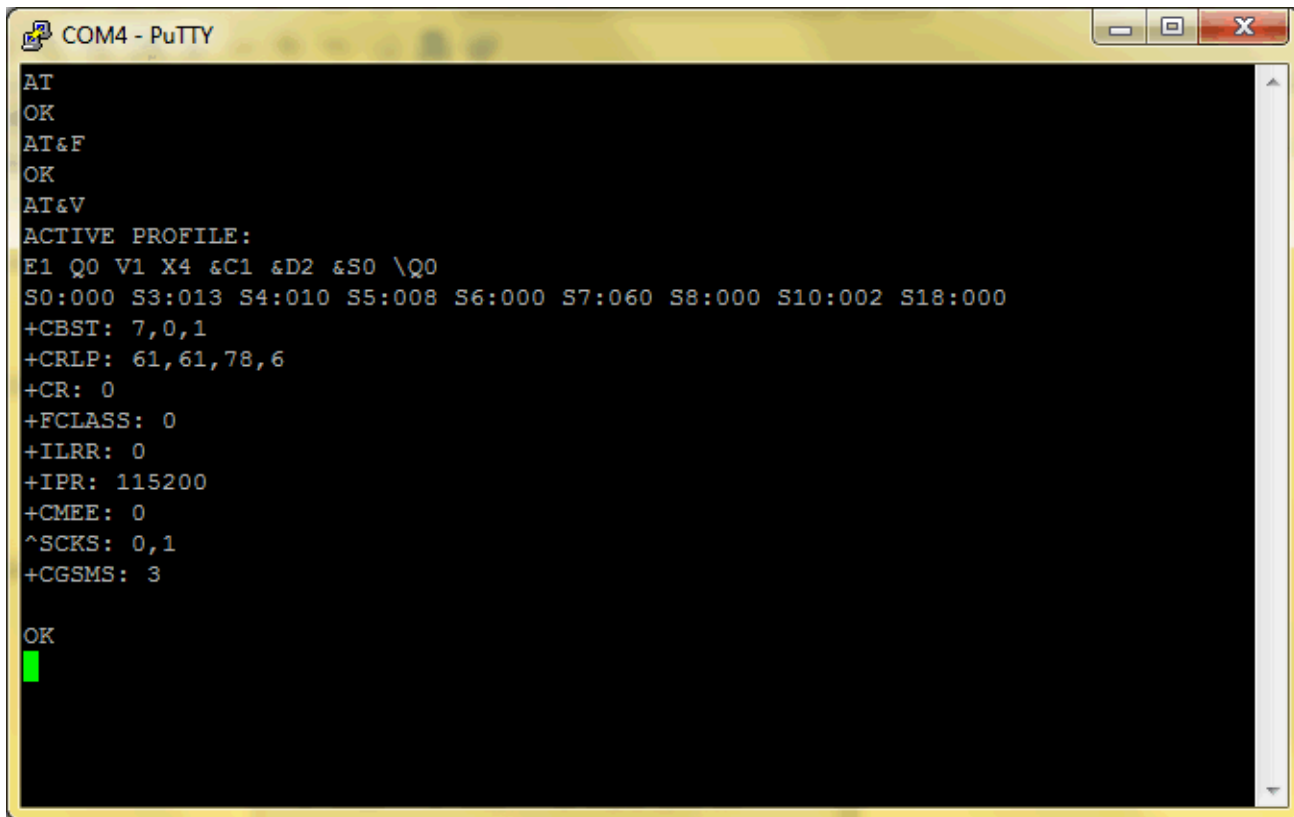
If there is no respons from the modem you should verify your settings.

Reset the modem to the factory defaults

The AT&F command resets all modem parameters to the manufacturer defined profile. The modem response should be OK.
The AT&V command lists the current parameters. Consult the Cinterion AT command guide for more info.
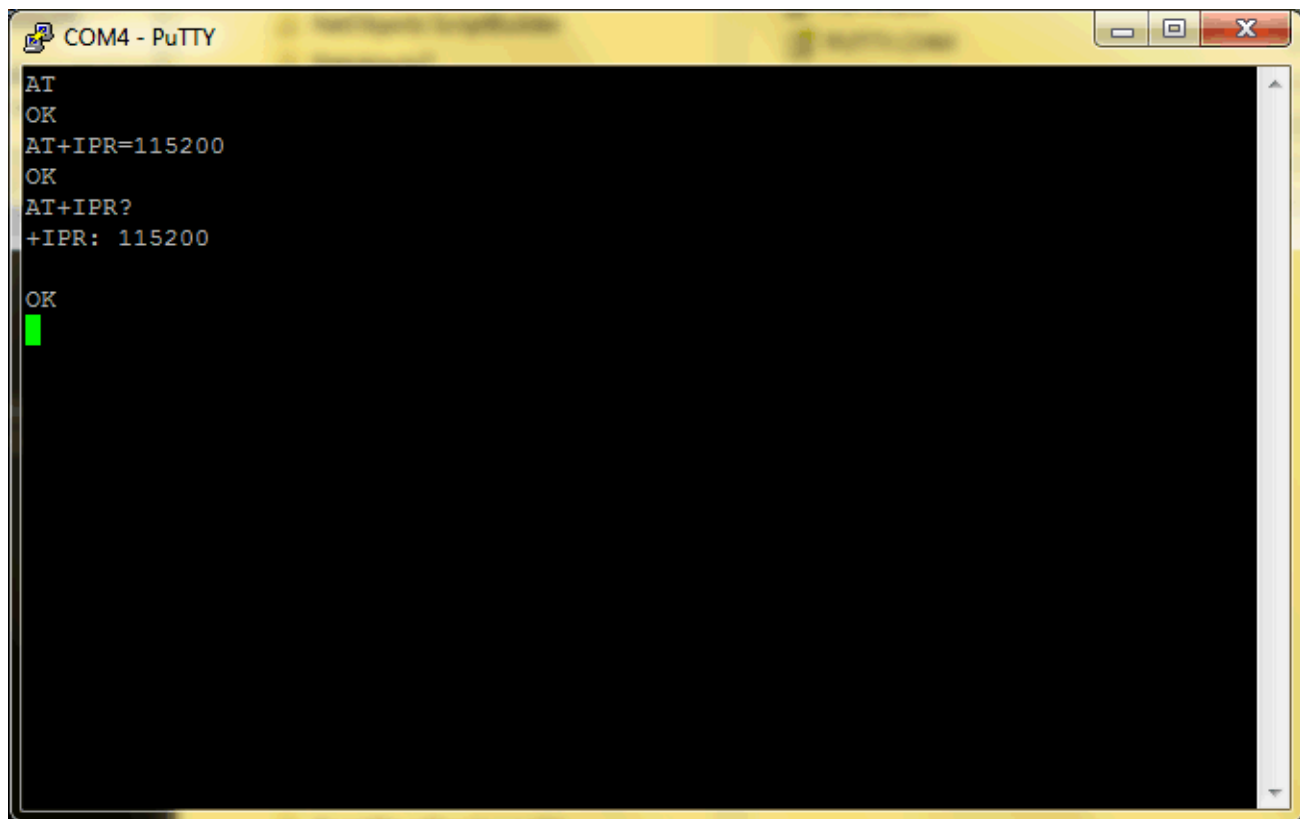
```
COM4 - PuTTY
AT
OK                                        - 23 -
AT&F
OK
AT&V
ACTIVE PROFILE:
E1 Q0 V1 X4 &C1 &D2 &S0 \Q0
S0:000 S3:013 S4:010 S5:008 S6:000 S7:060 S8:000 S10:002 S18:000
+CBST: 7,0,1
+CRLP: 61,61,78,6
+CR: 0
+FCLASS: 0
+ILRR: 0
+IPR: 115200
+CMEE: 0
^SCKS: 0,1
+CGSMS: 3

OK
```

Set the required modem parameters

The modem communication will be more reliable if autobauding is not used. The bit rate is specified by the

   AT+IPR=<rate>

command. Values of <rate> are 0 for autobauding, 300, 600, ..., 57600, 115200, 230400.
The setting will <u>not</u> be changed by the above AT&F command.
The current bit rate may be queried by the AT+IPR? command
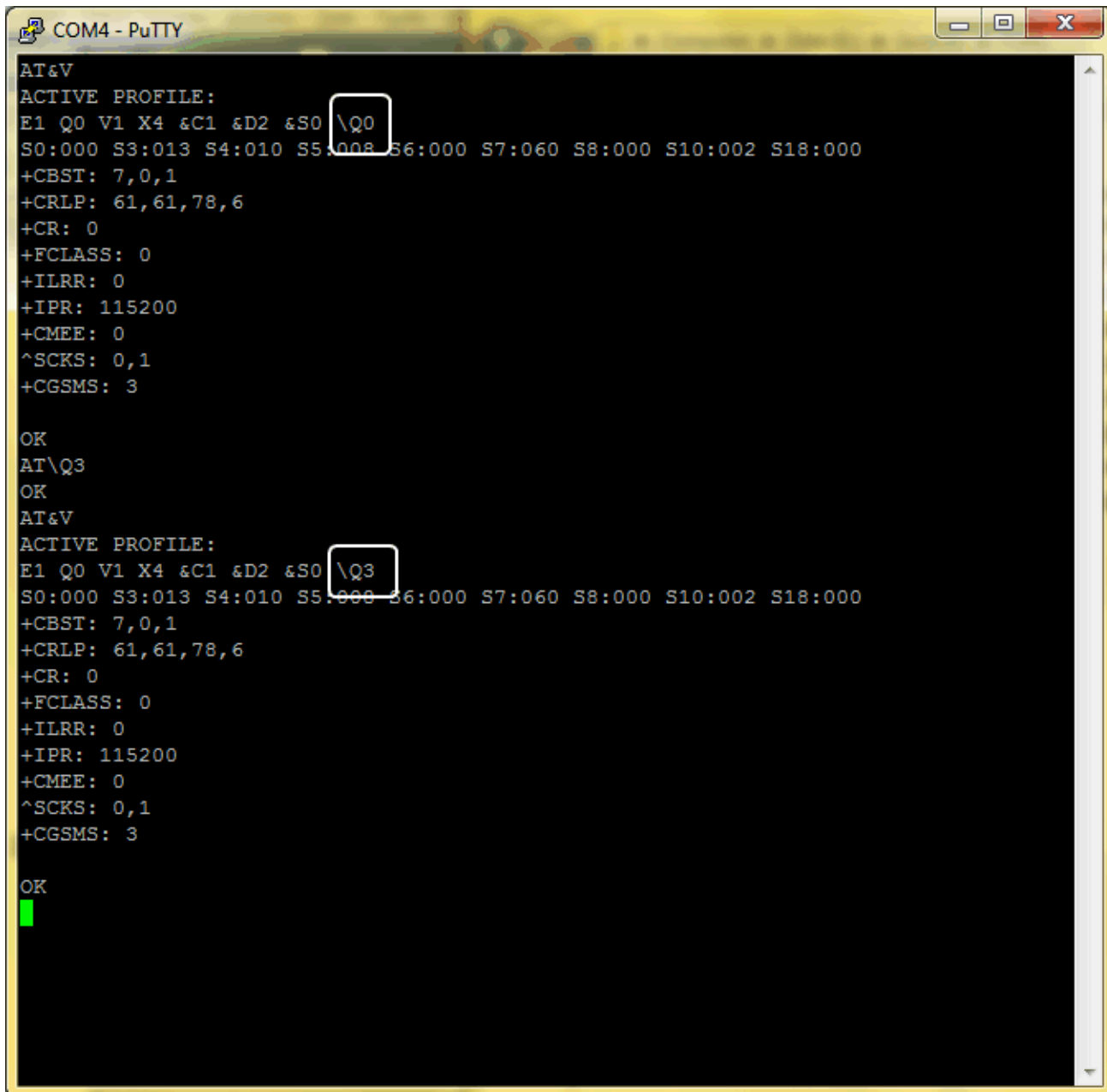
```
COM4 - PuTTY
AT
OK                                         - 24 -
AT+IPR=115200
OK
AT+IPR?
+IPR: 115200

OK
```

Flow control between the computer and the modem must be set to hardware flow control (using RTS/CTS).

Hardware flow control must be set on both sides: with AT\Q3 in the ME and an equivalent RTS/CTS handshake option in the host application.

The default setting of the modem is AT\Q0 (no flow control) which must be altered to AT\Q3 (RTS/CTS hardware handshake on). The setting is stored volatile and must be restored each time after rebooting the ME.

AT\Q has no read command. To verify the current setting of AT\Q, simply check the settings of the active profile with AT&V.

```
COM4 - PuTTY

AT&V
ACTIVE PROFILE:
E1 Q0 V1 X4 &C1 &D2 &S0 \Q0
S0:000 S3:013 S4:010 S5:008 S6:000 S7:060 S8:000 S10:002 S18:000
+CBST: 7,0,1
+CRLP: 61,61,78,6
+CR: 0
+FCLASS: 0
+ILRR: 0
+IPR: 115200
+CMEE: 0
^SCKS: 0,1
+CGSMS: 3

OK
AT\Q3
OK
AT&V
ACTIVE PROFILE:
E1 Q0 V1 X4 &C1 &D2 &S0 \Q3
S0:000 S3:013 S4:010 S5:000 S6:000 S7:060 S8:000 S10:002 S18:000
+CBST: 7,0,1
+CRLP: 61,61,78,6
+CR: 0
+FCLASS: 0
+ILRR: 0
+IPR: 115200
+CMEE: 0
^SCKS: 0,1
+CGSMS: 3

OK
```

Later, when configuring the Sms4Domino environment and gateways, you should define the same parameter settings.

Save the settings to non volatile RAM

The AT&W command is used to store the settings to non volatile RAM. The modem responds with OK.
Sms4Domino will issue the ATZ command as one of the first commands to load the settings into memory.
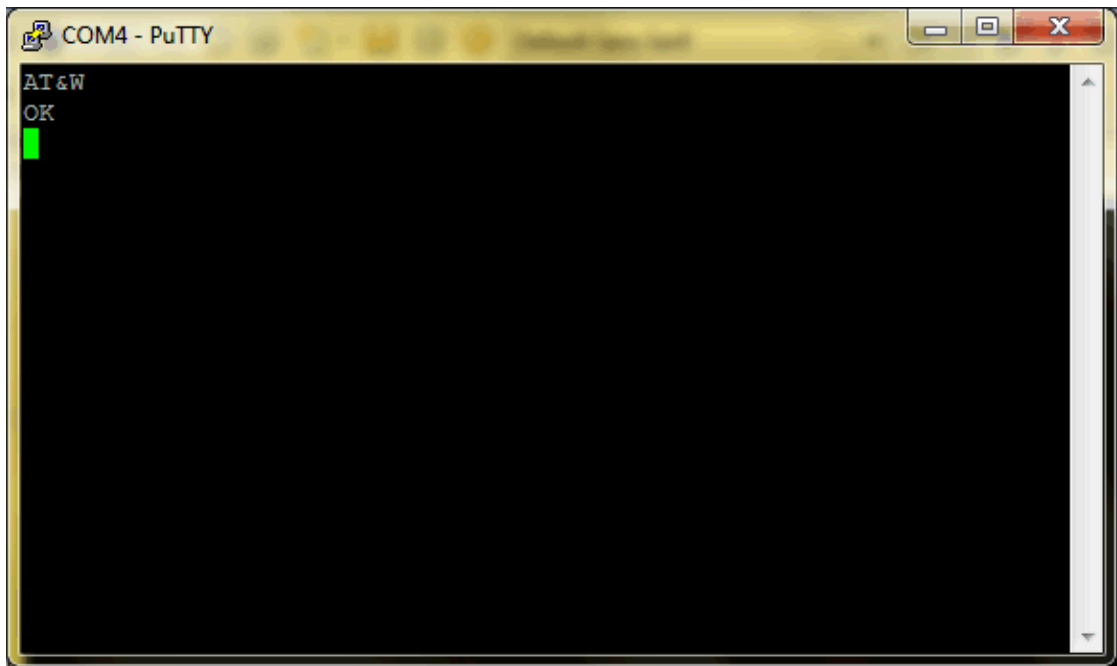
Verify the PIN code of the SIM card

Power down the modem. Insert a valid SIM card. You should have received a corresponding PIN code.

The AT+CPIN write command can be used to enter one of the passwords listed below. The read command can be used to check whether or not the ME is waiting for a password, or which type of password is required. This may be for example the SIM PIN1 to register to the GSM network, or the SIM PUK1 to replace a disabled SIM PIN1 with a new one, or the PH-SIM PIN if the client has taken precautions for preventing damage in the event of loss or theft etc. If requested by the ME AT+CPIN may also be used for the SIM PIN2 or SIM PUK2.
If no PIN1 request is pending (for example if PIN1 authentication has been done and the same PIN1 is entered again) MC52i responds "+CME ERROR: operation not allowed"; no further action is required.
Each time a password is entered with AT+CPIN the module starts reading data from the SIM. The duration of reading varies with the SIM card. This may cause a delay of several seconds before all commands which need access to SIM data are effective.
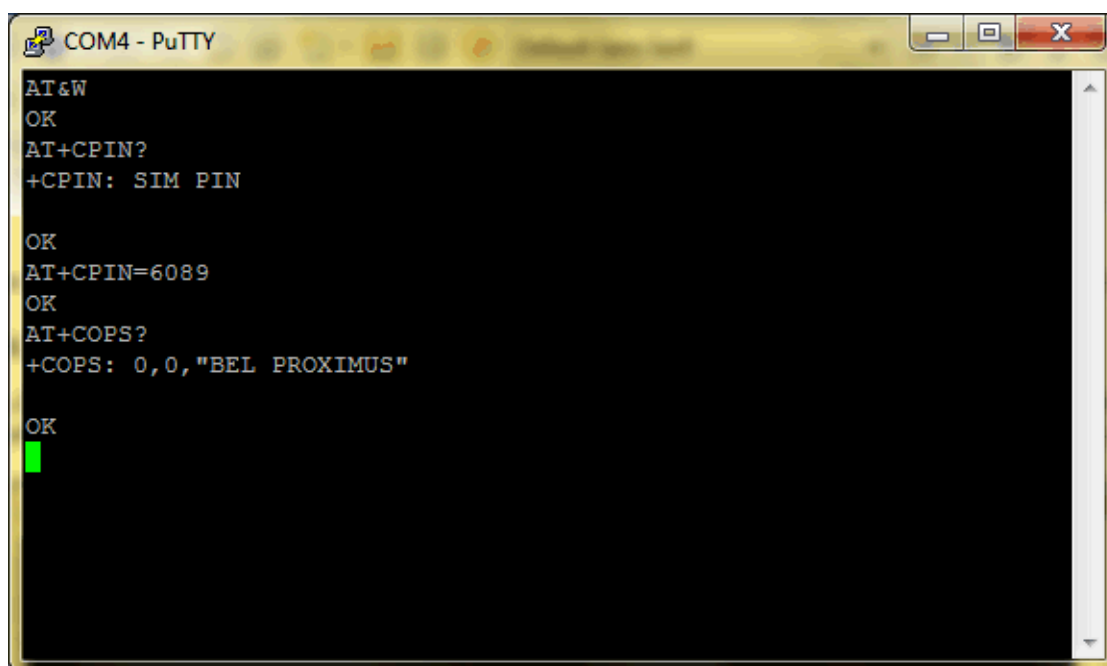
We could further verify that the modem has been properly registered to the GSM network. Enter the

AT+COPS?

command.

Appendix: Cinterion AT command guide

mc52i_atcommands_v01201a.pdf

Configuration of the ATOP SE 5000 Serial server

In case no COM ports are available or when the GSM modem should be located outside the computer room it possible to connect the modem to a serial server. This device acts as a gateway between the Ethernet LAN and the COM modems. The configuration of the serial server is documented in the user guide.

The current help document guides you through the setup. Following steps must be executed

- Connect the serial server to the LAN
- Assign a fixed IP address to the serial server
- Configure the serial server
- Save the settings
- Test the setup via a Telnet connection

Connect the serial server to the LAN

The serial server has following factory settings

IP address: 10.0.50.100

Subnet mask: 255.255.0.0

Default gateway: 10.0.0.254

DHCP client: disabled

User name: admin

Password: <null> (leave the password field blank)

Use a paperclip to push the reset button for a few seconds. You will hear some beeps. This resets the serial server to the factory default settings.

Configure the Ethernet port of your (test) computer with compatible IP settings. E.g.
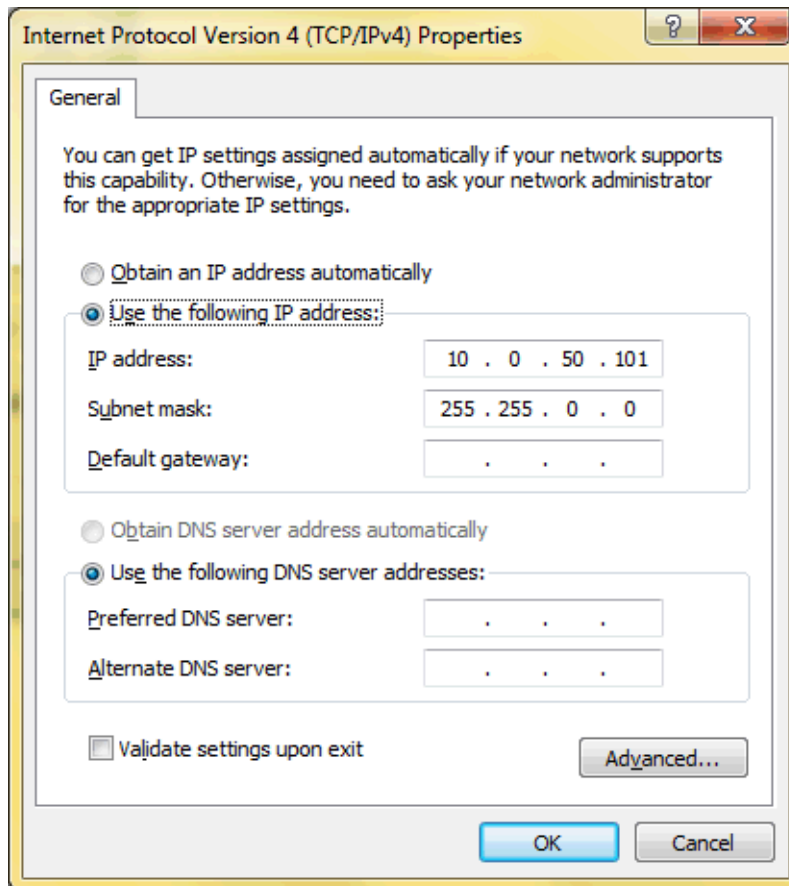
Connect the serial server to your computer. You need two straight cables and a switch or you could use a crossed cable and make a direct connecten between you PC and the serial server. Test the connection by ping-ing the serial server.
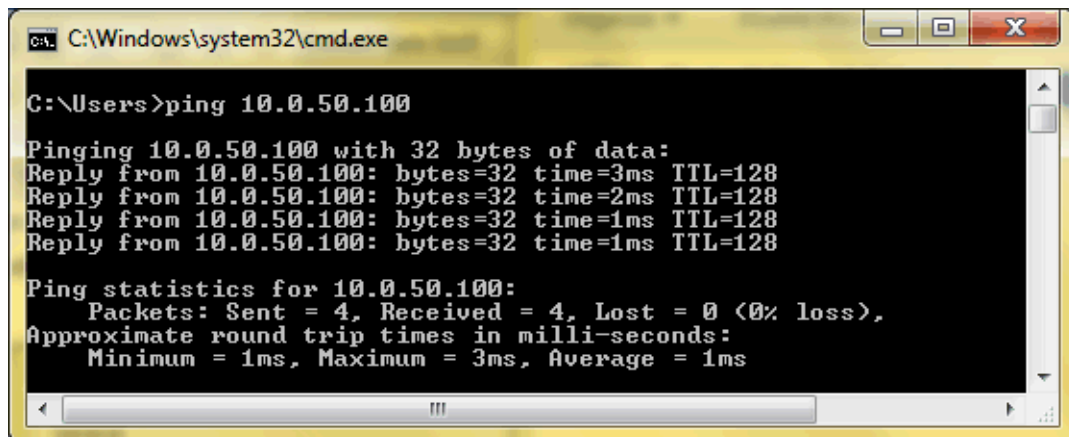
The serial server may be configured via the command line or using a browser. Internet Explorer should be used.
Enter http://10.0.50.100 as Url.

Assign a fixed IP address to the Serial Server

Ask your network administrator for a free IP address that is compatible with your network addressing scheme. The address may be assigned via DHCP or may be entered as a fixed address in the device. Sms4Domino requires the knowledge of this address.

Enter the new address in the *Networking* section
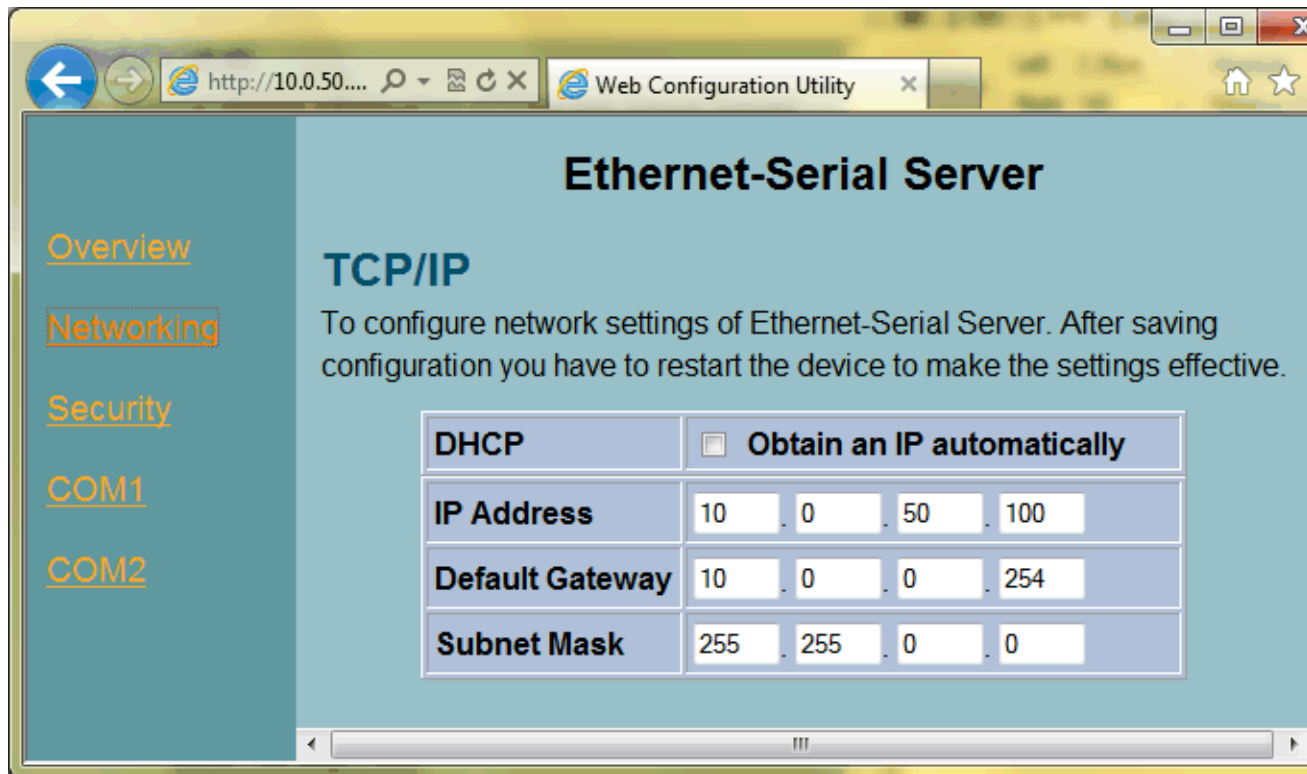
Click on *Save Configuration* followed by *Restart.*
You are now ready to connect the serial server to the LAN. Further configuration steps must be using the above assigned IP address.

Configure the serial server

Following instructions must be repeated for each of the COM ports. In the example we will configure COM1.
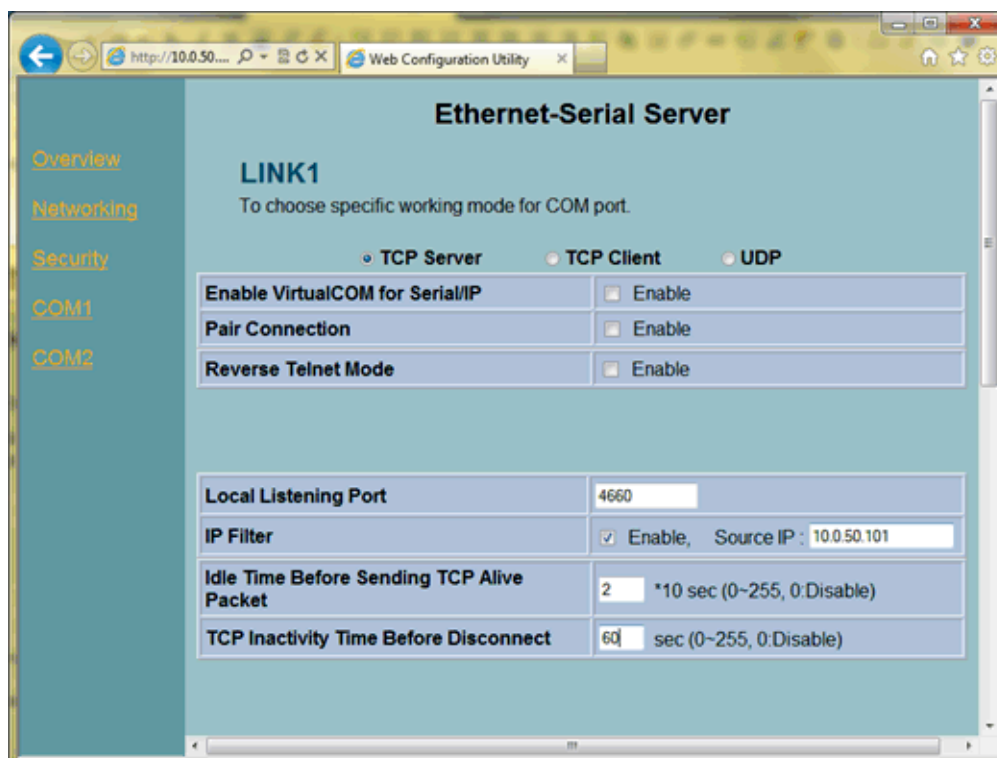
Sms4Domino supports TCP/IP connected modems. There is no need for special driver software that must be installed in Windows.
The serial server must be configured with following settings

TCP Server
Enable VirtualCOM for Serial/IP: No
Pair Connection: No
Reverse Telnet Mode: No
Local Listening Port: 4660 or any other value. The same value must be specified in the Sms4Domino gateway definition document
IP filter: this is a security feature. Only the Sms4Domino server should be allowed to communicate with the serial server.
   Enabled: Yes
   Source IP: specify the IP address of the Sms4Domino server
Idle Time before sending TCP Alive packet: Sms4Domino assumes the TCP/IP connection remains open. Enter 2 for 20 seconds
TCP Inactivity Time: in case of a failure of Sms4Domino the connection should be broken. Specify 60. This value should be higher than the watchdog interval specified in the application profile (SmsLib settings)

Following set of parameters correspond to the physical COM interface. The settings must be the same as the ones that have been configured in the modem.

Alias Name: for your own documentation
Baud rate: 57 600 (this is the default for the Cinterion modem)
Parity: none. (this is the default for the Cinterion modem)
Data bits: 8. (this is the default for the Cinterion modem)
Stop bits: 1 bit. (this is the default for the Cinterion modem)
Flow control: RTC/CTS (this is the default for the Cinterion modem)
Empty Serial Buffer: Yes
Data Packet Delimiter: this parameters determines at what moment data received from the modem should be transmitted to Sms4Domino. There is no fixed terminator character that applies to all responses. The only acceptable data packet delimiter is based on the inter-character time gap. A value of 15 msec should be fine.

COM Type Selection: RS232

Click the *Save* button to save the settings.

<u>Test the connection</u>

We assume the modem has already been configured. Connect the modem to the COM port of the serial server.

Open a Dos box and enter the command (assuming the serial server is configured for IP address 10.0.50.100 and to listen on port 4660)
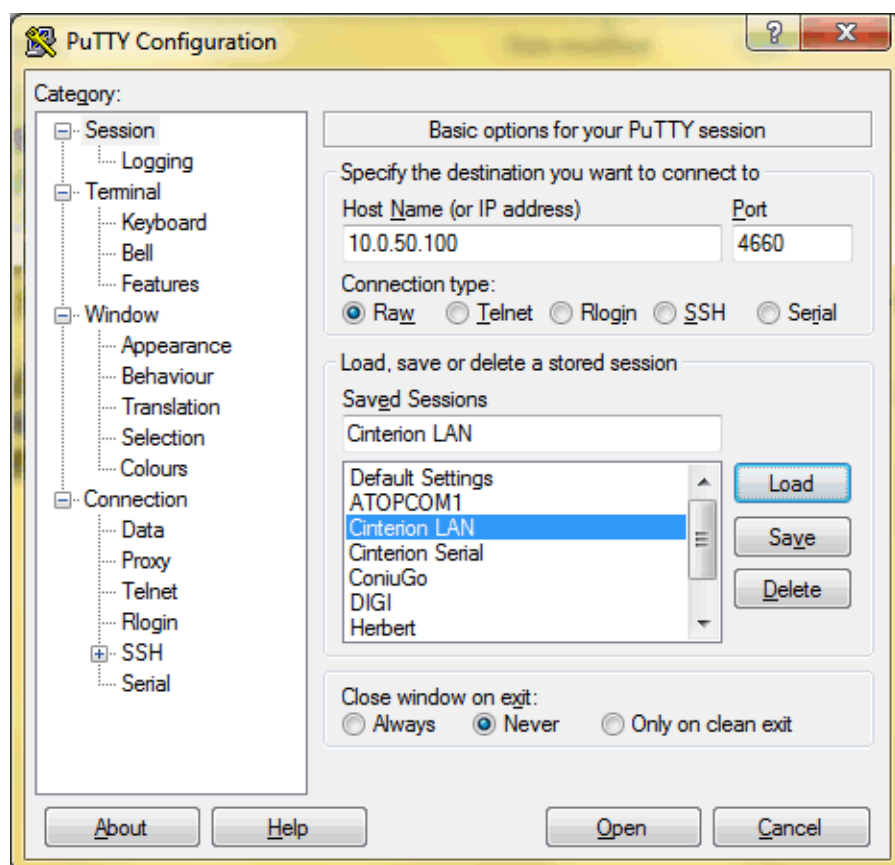
telnet 10.0.50.100 4660

The Dos window is cleared. Type AT. The modem should respond with OK
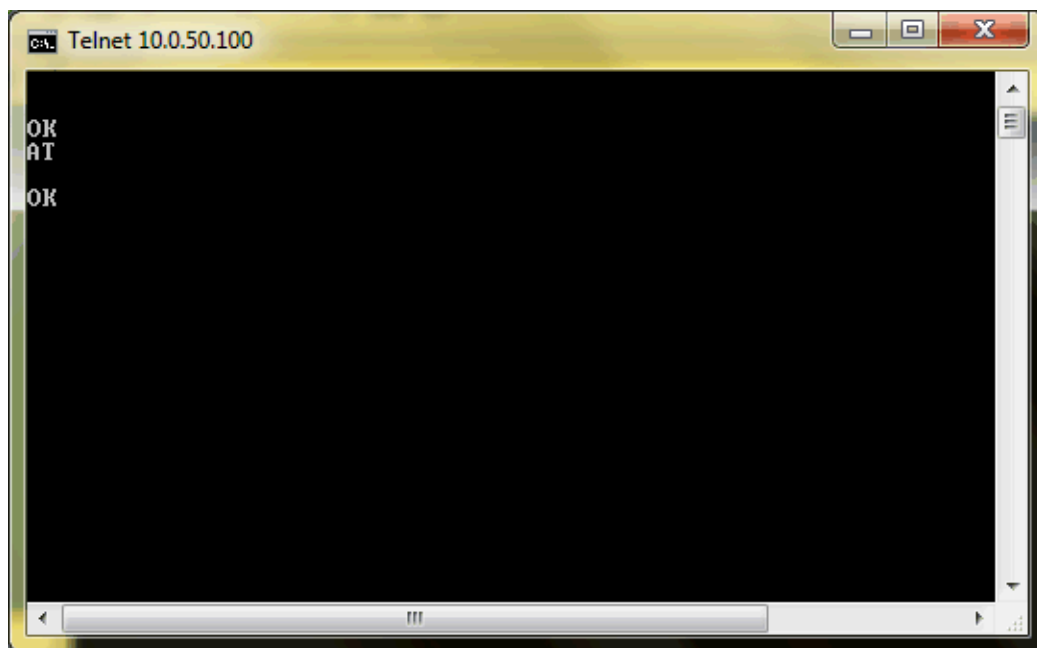
After the configured idle time the connection is closed automatically.

The same test could be performed using Putty. The connection type must be set to Raw.

Appendix: Atop user guide

SE5002 Serial- Ethernet User's Manual (V1.3).pdf

Configuration of Nokia GSM modem

An evaluation version of Sms4Domino may be downloaded. During that period you could consider using a GSM (mobilephone) for sending and receiving Sms messages. We strongly recommend to obtain an industrial grade modem when moving into production.

In this section we will explain how to connect a Nokia GSM to Sms4Domino. More detailed information is available at Nokia's web site. Connection of other modems will be similar.

Configuration steps

- Verify that your GSM has been tested with Smslib. Do not proceed if your phone is not listed. Consult the FAQ section in this help file for a complete list. In following example we will use a Nokia 6070 phone.

- Install a SIM card in the GSM. Power on the GSM and supply the PIN code. Try to send/receive at least one SMS message.

- Obtain a compatible cable to connect your phone to a USB port of your computer. In following example we will use a Nokia CA-42 cable. In theory you could use a Bluetooth or infrared connection. However we experienced fewer difficulties when using a cable.

- Download the driver software. In case of Nokia it is not required to download and install the full PC suite. Only the cable driver must be installed.

- Install the cable driver software before connecting the GSM to the PC. Consult the attached quick installation guide for more details.

- Connect the GSM to the computer.

- Verify the virtual COM port that has been created. Open the Windows control panel, and select System -> Device manager. You should find the GSM modem within the Modems category (Nokia CA-42 USB modem in the example)
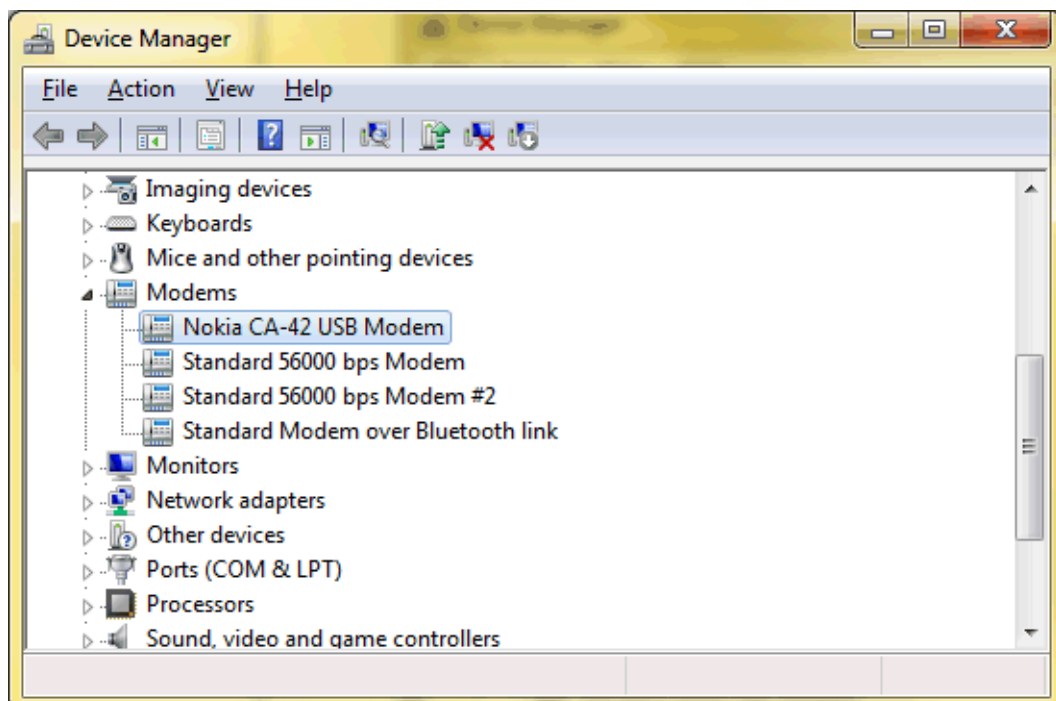
- Double click the entry for the Nokia modem. The port number (e.g. COM6) is listed within the Modem tab. You will need this number when configuring the Sms4Domino gateway document. You may need to lower the maximum port speed. The speed used by Sms4Domino will be defined later in the Sms4Domino gateway document.
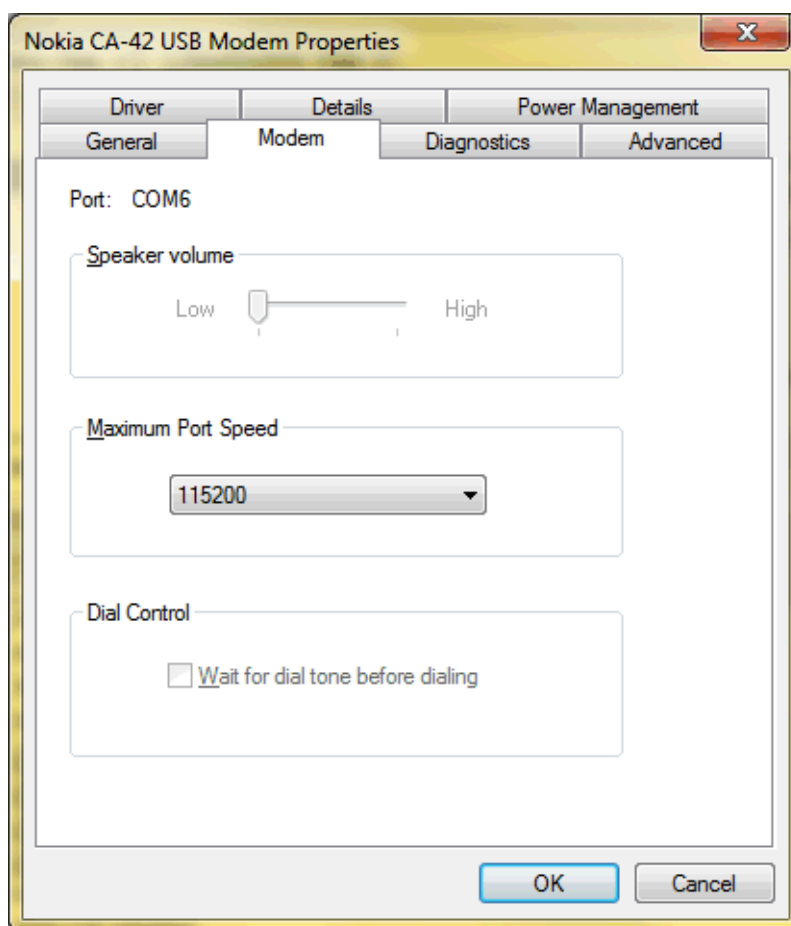
- Test the communication with the GSM. Click on the Diagnostics tab and click the Query Modem button. In this example the Nokia 6070 was identified. In case you get no response you could lower the communication speed with the modem.
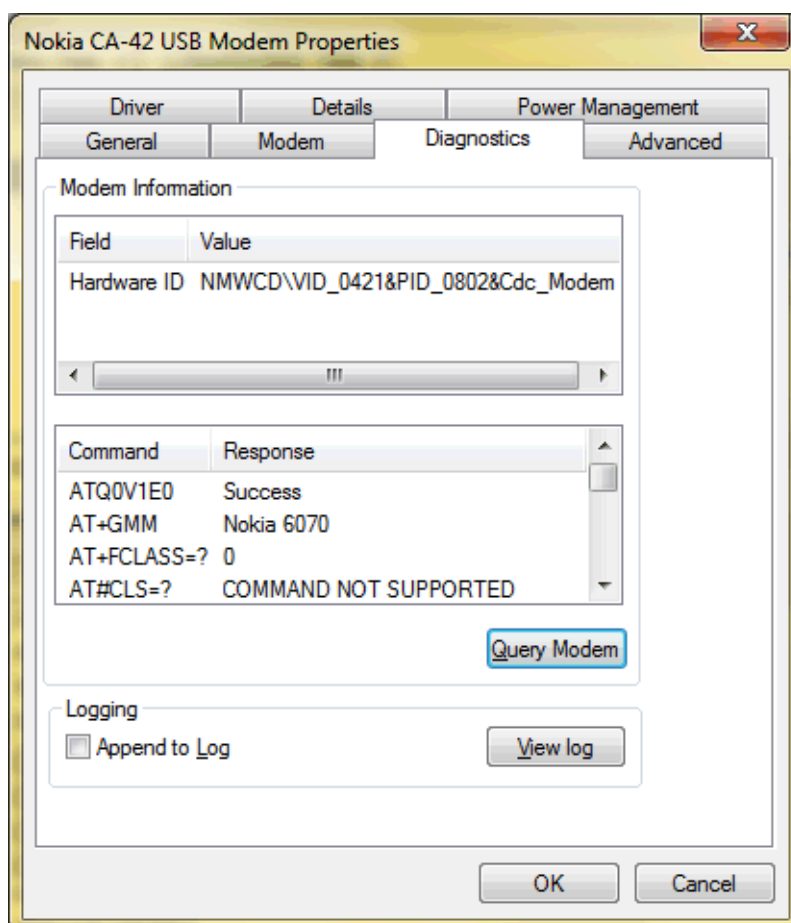
● The GSM is now ready to be used with Sms4Domino.

Additional resources

Nokia quick installation guide : Nokia_Conn_cable_driver_installation_eng.pdf (this info was found at Nokia's support site)

Clickatell Sms service provider

Sms service providers are an alternative to physical modems. In general they offer much higher throughput than a GSM modem. However, if you use Sms4Domino for sending alerts in case of a network failure, you should not use the network under test for sending Sms messages.

Sms service providers offer different API's for sending/receiving SMS messages. Smslib always uses the HTTP(S) interface and can only be used for sending SMS messages.

In order to use this kind of infrastructure you need to register with the service provider and purchase credits (some offer free credits for evaluation purposes).

In this document we will give a short description how to work with Clickatell. More detailed info is available at http://www.clickatell.com

Preparation steps

- Register for an account at http://www.clickatell.com. Registration is free. Select Clickatell Central (api). Follow the instructions. You may receive a mail with an activation link.

- When the registration process has been completed you must have a username, a password and a client ID.

- Login to the web site

- Add the HTTP API connection to your list of products. You will receive an API-identifier. You will need this identifier during the configuration of the Sms4Domino gateway document

- Send a test SMS. Open a browser that has access to the Internet and enter following url

http://api.clickatell.com/http/sendmsg?user=USERNAME&password=PASSWORD&api_id=APIID&to=PHONE&text=MESSAGE

(of course you should replace USERNAME, PASSWORD, APIID, PHONE, MESSAGE with valid parameters). The destination phone number must be written as an international numer without the leading + or without the leading 00 (e.g. specify 32474963333 for a number in Belgium).

● Within a minute you should receive the Sms.

Additional resources

Clickatell HTTP API guide: Clickatell_HTTP.pdf . The first pages describe how to register and how to obtain the API identifier. The guide was downloaded from Clikatells web site.

Decide on the architecture

Sms4Domino may be deployed in different configurations. However the program, as the name implies, always require the presence of a Domino server. This document describes a single instance configuration.

Running the Sms4Domino program as a server add-in

Sms4Domino is a Java program. The program should run continuously. As Domino agents are limited in execution time it is not possible to run the program as an agent.

Sms4Domino may run as a Domino add-in task. The program is started via the *load runjava Sms4DominoAddin* console command (case sensitive). All input/output between the Domino administrator and the program goes via the Domino (remote) console. As a shortcut the program may also be started using the *load runjava Sms* console command.

In order to run a Java program as a server add-in command Sms4Domino makes use of the open source software JAddin (http://abdata.ch/JAddIn.html). This kind of integration has not been certified by IBM®. Till today we have not yet experienced any problems.
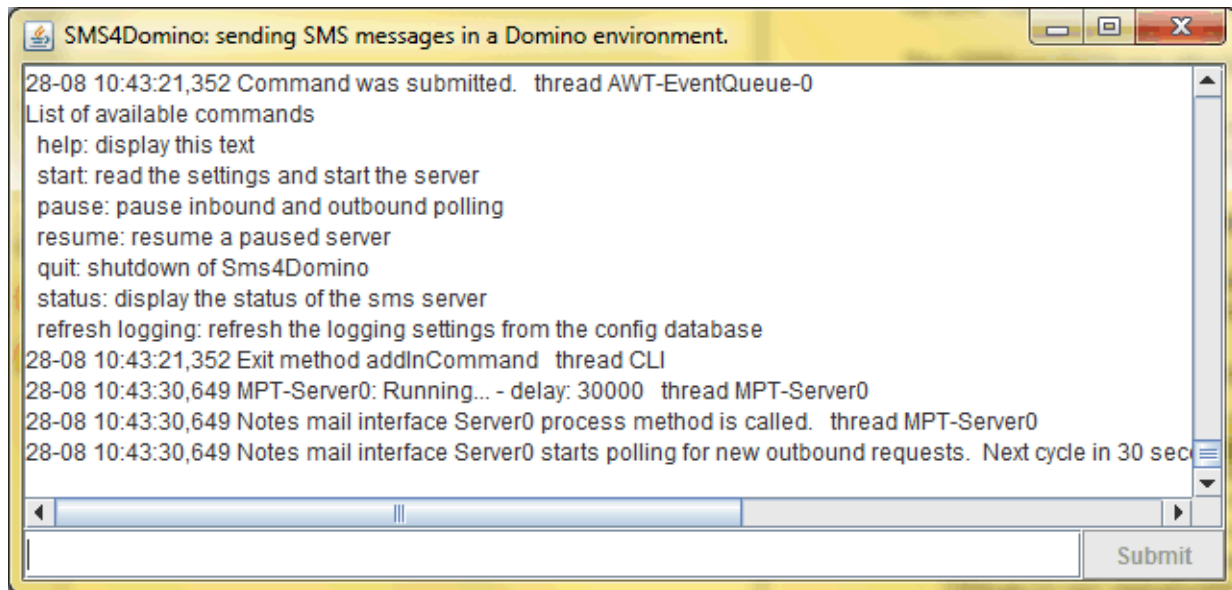
The GSM modems are attached to the Domino server.

The main advantage of this configuration is the integration of the input/output with the Domino console and log.nsf file. It is possible to use the Domino remote console to interact with the program. The program may be started using Domino program documents or notes.ini settings.

Running the Sms4Domino program as a standalone program

Sms4Domino may run as a stand-alone program on a workstation. Lotus Notes® must have been properly installed and configured on the workstation. Sms4Domino will access the Domino server using the credentials of the current Lotus Notes® user-id (defined in notes.ini by the KeyFileName variable). The password must be supplied as a start-up parameter or the user id must have no password.
Sms4Domino will create a console for interacting with the administrator.

The GSM modems are attached to the workstation. The configuration database must be installed on the Domino server.

This set-up does not depend on the JAddin program. However there is no integration with the Domino logging, no possibility to use the remote Domino console and supplying the Notes password via the command line may be less than ideal. In case the Domino server is stopped or the network connection becomes broken, Sms4Domino will not be able to continue, will log errors to the console and, in most cases, must be restarted in order to recover from the failure.

Databases

The main database is the configuration database. It contains the application profile, the instance configuration documents, the license file and the configuration documents for the gateways and the interfaces. The database is based on the Sms4Domino template. The database must be located on a Domino server and not on a workstation.

In most cases requests for sending Sms messages are created via Lotus Notes® mail messages (received by the *Notes mail messages* interface). The Domino routing configuration requires the definition of a foreign domain document. The foreign domain documents defines a gateway file name. This could be the configuration database or a separate database. The database should always be based on the Sms4Domino template. We recommend using a separate database.
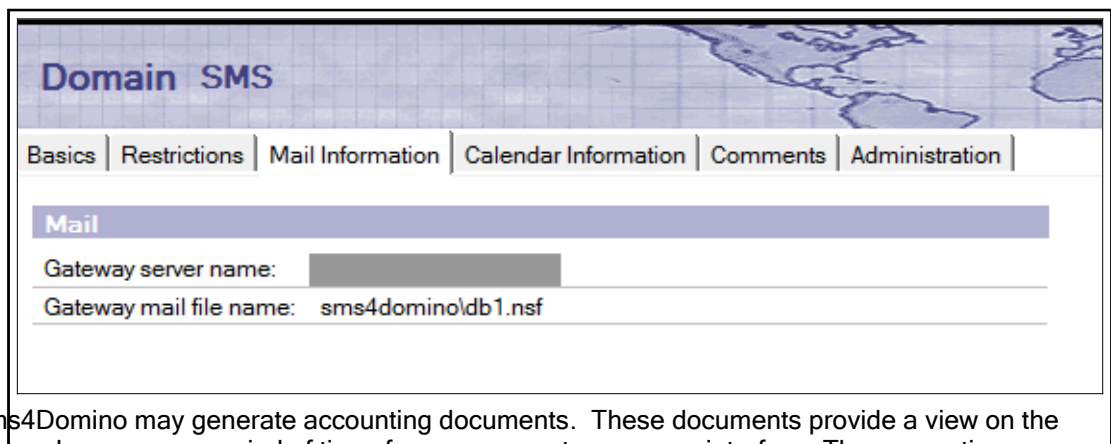
Sms4Domino may generate accounting documents. These documents provide a view on the Sms volume, over a period of time, for a user, a gateway or an interface. The accounting documents are part of the control database.
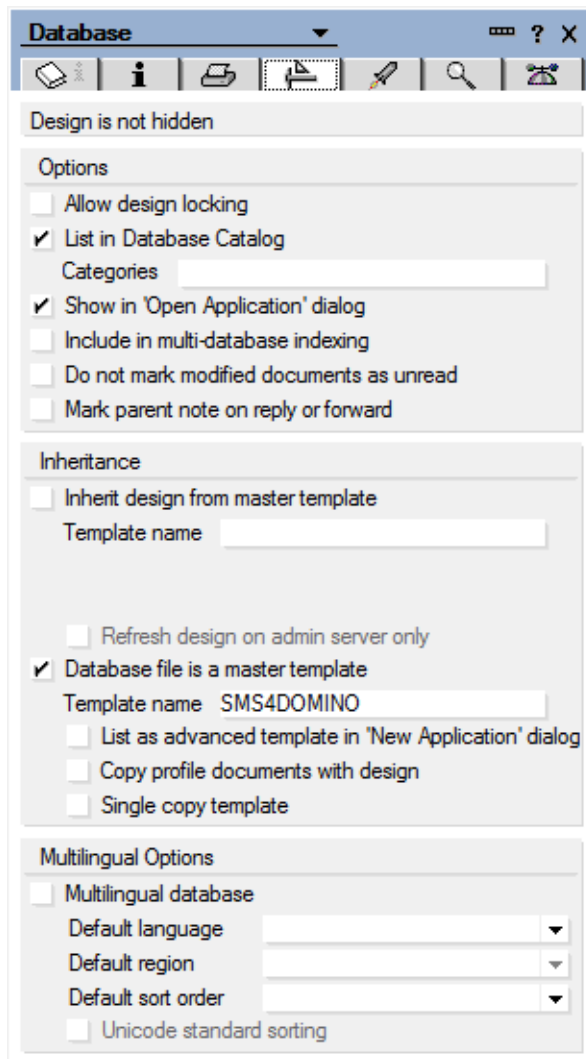
Installation of the databases

TheSms4Domino software is shipped as a set of two Domino databases : the Sms4Domino template and the help file. The template is a common template that must be used for all Sms4Domino related databases.

The files may be delivered on a CD, by e-mail or may be downloaded from the Internet. The Sms4Domino template database is distributed as a Notes template file (.ntf). The help database is distributed as a standard Notes database file (.nsf).

The databases may only be accessed by a Notes client and not by a web browser.

The current version of the template is 3.0.

- Get a copy of the databases. Unzip the files if required.

- Copy the template to your local Lotus Notes® data directory. By preference this should not be done on the Domino server but on a workstation of the administrator. Do not forget to remove the "read-only" attribute that may be present at the level of the operating system.

  The ACL settings of the template grants full access to all users. This may be changed according to your company policy settings.

- Sign the template using the Notes user-id of the administrator in order to avoid any future ECL warnings. This user-id should have sufficient access rights to run Lotusscript agents.

> You could be prompted for the creation of a cross-certificate in your address book. Click No.

- You may adapt the ACL of the template to conform to your local standards.

- Use the Notes client to copy the template to the Domino server. The template on the workstation is then no longer needed.

- Create a configuration database from this template. The database must be stored in the data directory or a subdirectory on the Domino server. The title and file can be any valid name. (In the help document *Notes.ini settings* you will learn how to tell Sms4Domino where to find this configuration database). The license file that is required to run Sms4Domino is based on the server name where the configuration file is located. Moving the configuration database to another server requires the presence of a new license file.
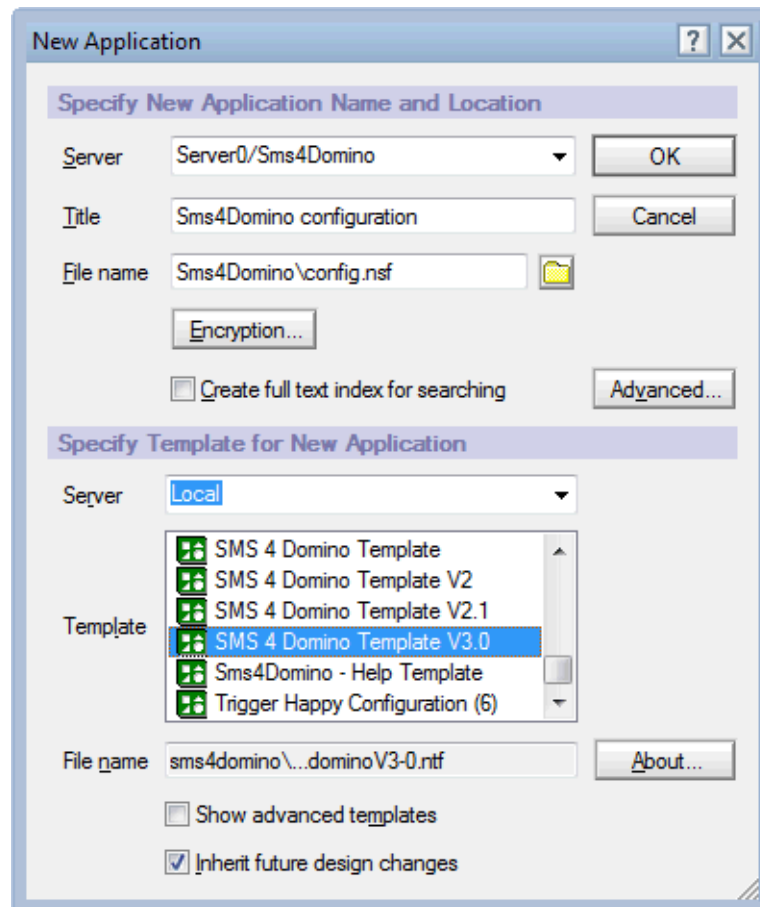
- Modify the default ACL of the database. Details of the ACL settings may be found in the document ACL Settings of the application.

- Some database properties for this database

  - When opened in the Notes client : Open designated frameset *Main1*

  - When opened in a browser : Open designated page *AccessForbidden*

  - Full text index is not required

- The database may be replicated to other Domino servers in case of a multi-instance configuration.

- Copy the help file to your local Lotus Notes® data directory. By preference this should not be done on the Domino server but on a workstation of the administrator. Do not forget to remove the "read-only" attribute that may be present at the level of the operating system.

- Sign the database using the Notes user-id of the administrator in order to avoid any future

ECL warnings.

● Use the Notes client to copy this database to the Domino server. The copy must be stored in the data directory or a subdirectory on the Domino server. The file name can be any valid name. The database on the workstation may be deleted. (In the help document *Application profile settings -> General settings* you will learn how to make a link between the Sms4Domino configuration database and the help database).

● Modify the ACL of the database. The default ACL grants manager level access to all users. Standard users should have read only access.

● A full text index is recommended on the help database.

● In case separate databases are used for the accounting documents and/or the foreign domain mailbox the procedure for creating the configuration database must be repeated. The application profile in these databases contain a link to the main configuration database.

ACL Settings of the Sms 4Domino configuration database

This database is the central point for the application. It contains all configuration settings and eventually queues of Sms requests, accounting data, ...

This database should not be accessible by unauthorised users. It should get the same level of protection as the Lotus Domino® directory.

Access to the application is controlled by access levels and roles in the ACL.

The available roles and their purpose are explained in following table.

Role	Purpose
[DBAdmin]	- grants write access to the application profile - grants access to the run-time dll files - grants author level access to all configuration documents - this role is normally assigned to a single administrator, responsible for the set-up of the software - the owner of the role should have Author level access or higher in the ACL

Recommended ACL settings for the server

- Manager level access (at least editor level access is required)
- All roles enabled

Recommended ACL settings for administrators

- Author level access
- Delete documents

Recommended ACL settings for the user -id running Sms 4Domino

When running Sms4Domino as a server add-in this user-id is the server-id. When running Sms4Domino as a standalone program this user-id is the id specified by the notes.ini entry KeyFileName. This corresponds to the user-id that was used the last time Lotus Notes® was started.

- Manager level access (at least editor level access is required)
- All roles enabled

Recommended ACL settings for web users

- No access (the database has not been designed for web access)
- No roles enabled

Installation of the runtime files

Sms4Domino is a Java program that may run as a server add-in or as a standalone program. The program depends on some open source libraries (see the document *About this database* in the current help file). All code has been packaged in a few jar files *Sms4Domino.jar* and the jar files of the open source components.

Java must be able to communicate with the GSM modems over the serial COM ports. One dll file is required. There is a specific file for 32 bit Windows and another for 64 bit Windows.

All runtime files may be found in the configuration database via the menu *System -> Runtime files*.

You need the [DBAdmin] role in order to access the page containing the runtime files.

Procedure

1. Open the Sms4Domino Configuration database.

2. Select *System -> Runtime files* in the left navigation pane. A page containing the jar and dll files opens.

3. Open the *Windows 2003 - Windows 2008* bit section.

4. Detach all files into the specified directories on the machine where the Sms4Domino program will be running.

5. Close the window.
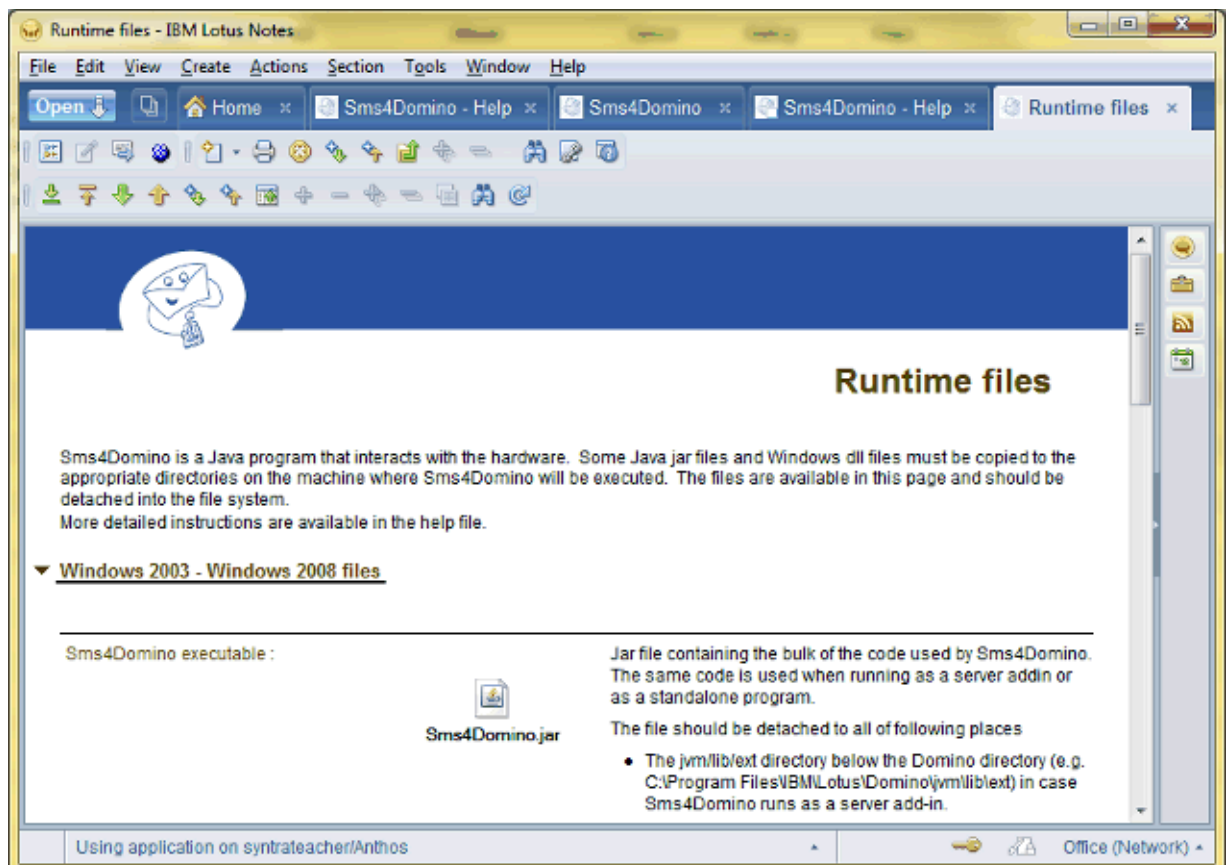
Notes.ini settings

Most of the run time parameters for the Sms4Domino software are specified in Lotus Notes® documents in the configuration database. However, when Sms4Domino is started, it must know where to find the configuration database.

This is achieved by two mandatory notes.ini settings. When Sms4Domino is running as a server add-in the notes.ini of the server must be modified. When Sms4Domino is running as a stand-alone program the notes.ini of the Notes client must be modified.

The notes.ini of the Domino server may be modified by editing the file directly. However, it is recommended to specify the settings in a configuration document in the Domino directory. The notes.ini of a workstation is edited using a plain text editor like Notepad.

The list of available notes.ini parameters is specified in following table. The parameters are not case-sensitive. Some of the parameters are mandatory !

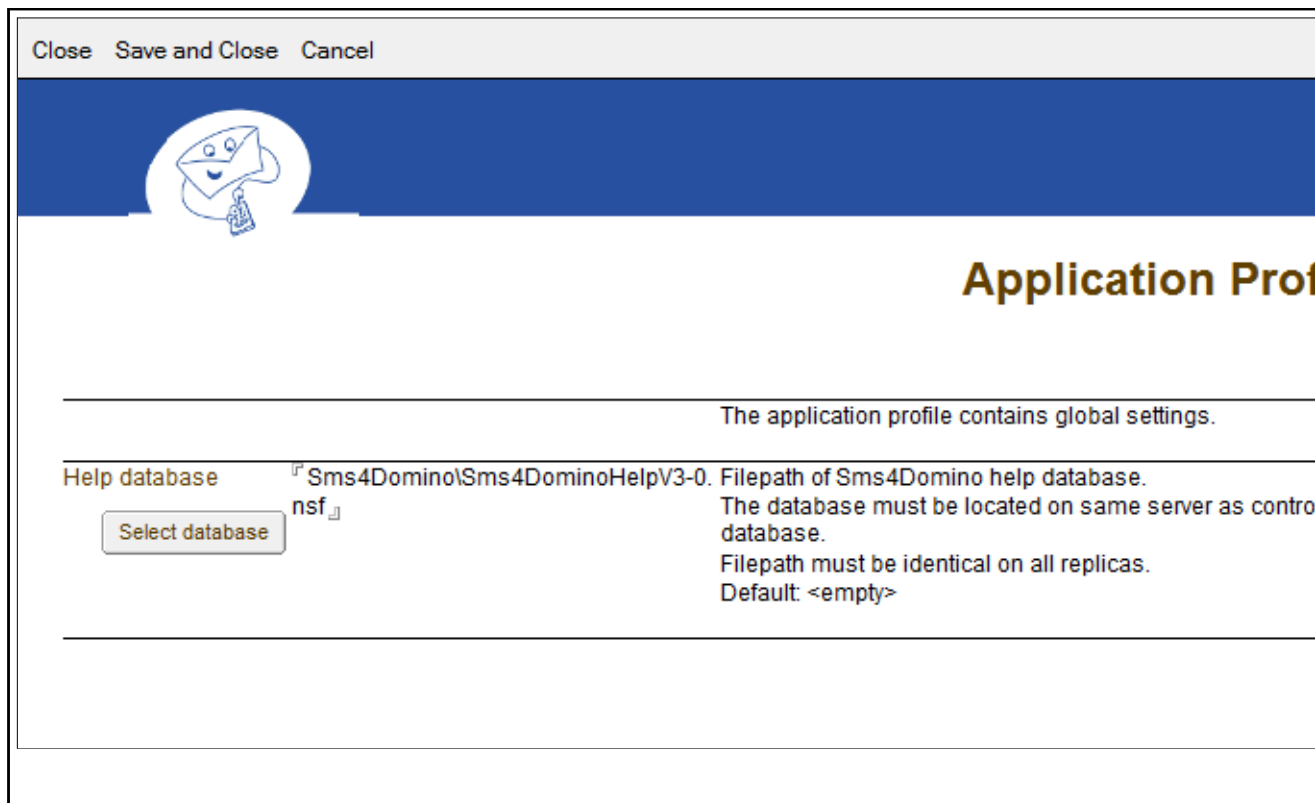| Parameter	Purpose
Sms4DominoConfigDB	This parameter specifies the location of the Sms4Domino configuration database. This database resides in the Domino data directory or one its subdirectories. The parameter must be specified as a relative file name. The parameter is case sensitive on some platforms (e.g. Unix) Example : Sms4DominoConfigDB=Sms4Domino\configdb.nsf Default value: sms4Dcfg.nsf
Sms4DominoServer	This parameter specifies the hierarchical name of the Domino server where the configuration database resides. In case the parameter is not specified the filter assumes the database resides on the current server. Example : Sms4DominoServer=Acme/SVR/Comp Validation of the license file is only possible if the configuration database is located on a Domino server.
Sms4DominoInstanceID	Every running Sms4Domino program must be identified by a unique hierarchical name: the instance ID. This name is used in most of the configuration documents. By default - when this parameter is not part of the notes.ini file - the instance ID is the same as the hierarchical Domino server name that holds the configuration database. Example: Sms4DominoInstanceId=Acme/SVR/Comp
Sms4DominoConsoleWidth Sms4DominoConsoleHeight	These two parameters are only used when Sms4Domino is running as a stand-alone program. They define the width and height, expressed in pixels, of the console used for logging and entering commands. The open console may be resized by the Windows user. When Sms4Domino is stopped the current console size is saved into these two environment variables. Example : Sms4DominoConsoleWidth=1000

	Sms4DominoConsoleHeight=500

Application profile settings

The application profile contains a few settings that are common to all Sms4Domino instances.

The profile document is accessed via the *System -> Application Parameters* entry in the menu outline. This document is mandatory and must be created by the database administrator. You need manager level access to create such a profile.

Procedure

1. Open the Sms4Domino configuration database.

2. Select *System -> Application Parameters* in the navigator. The profile document is opened in edit mode

3. Click on *Select Database* to specify the location of the help file. It is required that the help file is replicated together with the configuration database to all Domino servers that hold a copy of the confiuration database. The filepath should be identical on all Domino servers.

4. Click *Save and Close* to update the settings.

5. The control database must be closed and reopened

Close   Save and Close   Cancel

Application Prof

The application profile contains global settings.

Help database   「Sms4Domino\Sms4DominoHelpV3-0. Filepath of Sms4Domino help database.
nsf」   The database must be located on same server as contro
database.
Select database   Filepath must be identical on all replicas.
Default: <empty>

Instance configuration settings

Most of the instance related parameters of the Sms4Domino software are specified in the instance profile in the configuration database. There should be exactly one instance document per Sms4Domino instance.

It is required to stop and restart Sms4Domino in order to make the modifications to the instance configuration visible to Sms4Domino.

Every running Sms4Domino program is identified by a unique Instance ID. This ID is a hierarchical name. We recommend using the same name as the Domino server name where the corresponding configuration database is located. A running Sms4Domino program reads its Instance ID from the notes.ini file. More info in the help document *Notes.ini settings*.

Procedure

1. Open the Sms4Domino configuration database.

2. Select *System -> Instances* in the navigator. The right part of the screen displays all defined Sms4Domino instances.

3. Double-click on an existing instance document or click the action *New Instance Configuration*.

4. Assign a unique Instance Id to the document. Add a descriptive text for your own use.

5. Verify all default settings and modify them if necessary. Default settings may be restored at any time by clicking the *Tools* action button. More detail is provided in the following help pages.

6. Click *Save and Close* to update the settings.
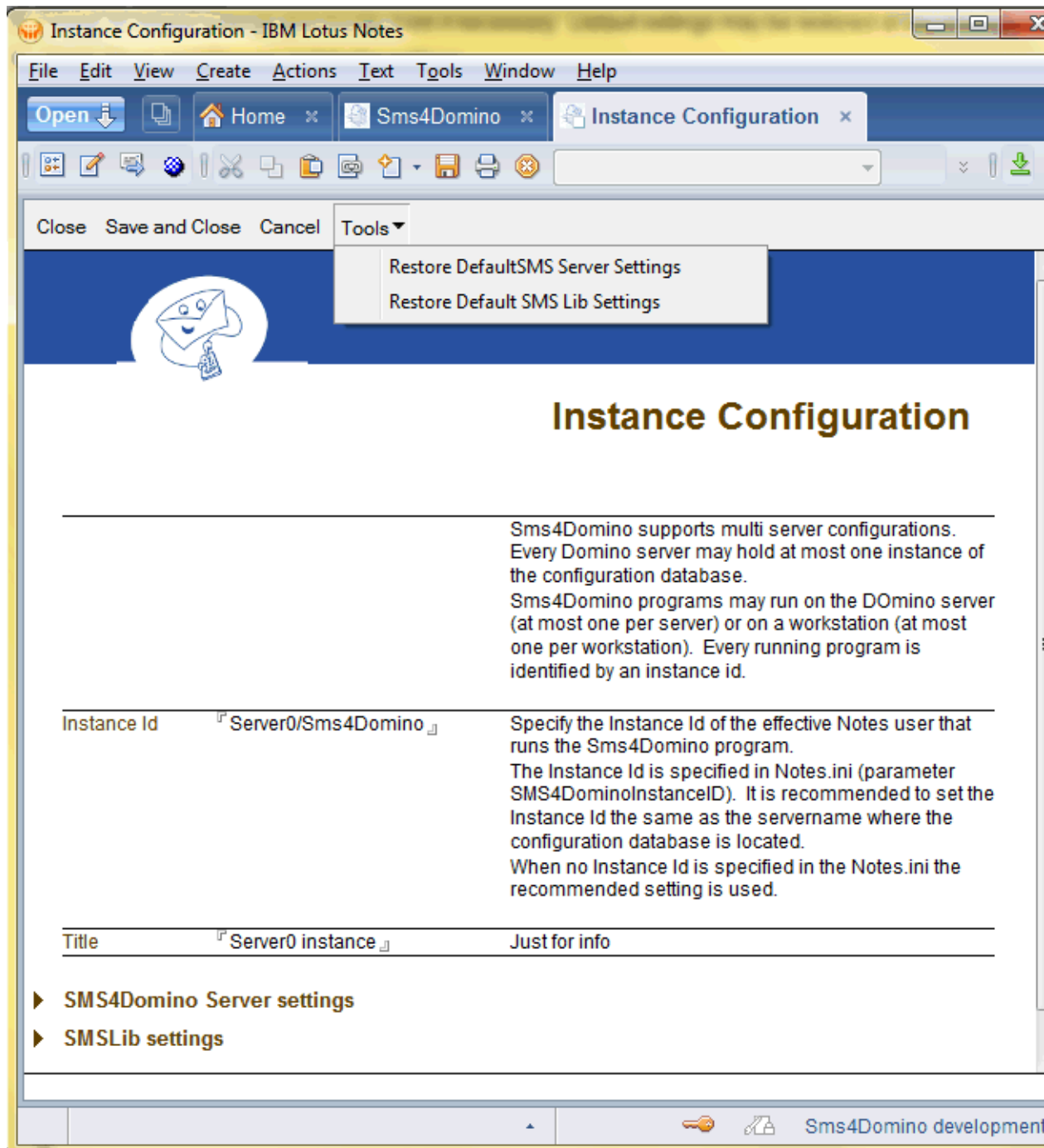
Sms4Domino Server Settings

This section contains global settings that are used by the Sms4Domino instance.

Parameter	Purpose
Autostart server	When Sms4Domino is launched, either as an addin or as a standalone program, it is ready to accept console commands. (The *Help* command displays a list of available commands). However the Sms server is not yet started by default. This allows the administrator to launch some test probes on the modems. By setting the Autostart server parameter to enabled the Sms server is started automatically as soon as Sms4Domino is launched. In production environments this should be the normal mode of operations. Values • Enabled: this is the recommended value • Disabled
Delete messages from GSM modem	GSM modems may keep a list of all outbound SMS messages in their internal memory. When this memory is full the modem may refuse to accept further messages. Sms4Domino may delete the Sms from the memory of the modem after transmission. Values • Enabled: this is the recommended value • Disabled. The same message will be received multiple times from the modem.
Outbound polling interval	The outbound direction is defined as *from Sms4Domino to the GSM network*; Requests for sending SMS messages may come from different sources. Every source is implemented by an interface. The interface places the request into an internal queue. Sms4Domino polls the queue every *outbound interval* seconds. Interfaces may accelerate the polling process by sending a signal to the polling process. As a consequence it is allowed to have a rather large outbound polling interval. Recommended value: 300 seconds
Outbound queue depth	Sms4Domino pushes outbound SMS requests into the queues managed by SmsLib. This parameter specifies how many outbound requests may be at any time in the SmsLib queues. In order to have an optimal throughput the queue should not be left empty when there are still outbound SMS requests present within the interfaces. Pushing too many requests into SmsLib may result in a memory overload. In case of a server crash the messages will be pushed again into the queue when Sms4Domino starts again. Recommended value: 2 * the number of outbound modems (outbound gateways)
No inbound route found	These settings specify how inbound SMS messages (received from GSM network) are processed in case no or multiple matching inbound routing documents have been found. • The received SMS may be forwarded by mail to 0, 1 or more valid mail

	addresses.
	● An alert may be mailed to 0, 1 or more valid mail addresses. The alert does not contain the text of the Sms. It is used to inform the Sms4Domino administrator that some unroutable SMS has been received.
	● Should the SMS be kept in the Sms4Domino database. Doing so gives the Sms4Domino administrator the opportunity to read and forward the SMS manually.
	● What should be the apparent sender of the mail that forward the received SMS or alert message. The apparent sender is composed of three or two parts
	<Name>@<Number>@<Domino domain name> or
	<Number>@<Domino domain name>
	The <Name> and <Domain domain name> must be specified. The <Domino domain name> must be configured in the Domino directory as a Foreign Domain. The <Number> will be extracted dynamicallay from the received SMS message. This allows the Notes user to reply via mail to the inbound SMS.
Heartbeat interval	Heartbeat documents are created dynamically by a running instance. A document created by Instance x informs all other instances that x is still running and provides an estimation of the current workload for x. The hearbeat documents replicate through the Sms4Domino network. Single instance configurations do not require regular updates of the heartbeat document. A negative value for this parameter results in the creation of a heartbeat document at program start.
	Recommended value: 60 seconds for multi-instance configurations. -1 for single instance configurations.
Hearbeat max age	When evaluating the health records of the Sms4Domino instances, the records that have not been updated since the specified number of seconds, are considered too old. The Sms4Domino instance will be considered as non operational. The specified value must be greater than the maximum heartbeat interval + the maximum time to replicate the heartbeat records through the network.
	A negative value disables the heartbeat check. This is allowed for single instance Sms4Domino architectures.
	Default value: 20 minutes
Remote instance notifier	Sms4Domino may perform least cost routing of requests for sending SMS messages (see Instance Routing Rules in this help file). When a mail is received by instance x, the instance calculates the best route and stores the route in the document. The document replicates to the other Domino server where it is picked up by instance y. In case of Domino clusters this replication occurs within a few seconds. If Domino clustering is not being used it could take too long to replicate the info from x to y. The Remote Instance Notifier on instance x may issue a Domino console command to server x. Typically this command will replicate the configuration database to instance y. The configuration of the Remote Instance Notifier consists of a few parameters
Minimum delay between commands	In order to avoid overloading the server a minimum delay between consecutive commands may be specified.
	Default value: 5 seconds.
Minimum delay	Similar to parameter *Min delay between commands*. In this case the

between commands for the same target instance	minimum delay between two commands for the same Sms4Domino remote instance is specified. Default value: 10 seconds.

Console commands	When the current Sms4Domino instance x performs least cost routing to another Instance y (target instance) all listed console commands will be issued on the Domino server that holds the configuration database of the current instance.
	This is a multi-valued field. Separate values by *newline*.
	Each line is formatted as
	   Target Instance ID%%Console command
	E.g.
	   Instancey/Sms4Domino%%replicate Instancey/Sms4Domino
	   sms4domino\config.nsf
	   Instancez/Sms4Domino%%replicate Instancez/Sms4Domino
	   sms4domino\config.nsf
	
	There could be zero, one or more commands for any target Instance ID

Low priority time ranges	Low priority SMS messages are sent after high and normal priority messages but may be sent at any time during the day. However, low priority messages are only given to the SMS server during the specified time ranges. Each time range is formatted as *starttime - end time*. Multiple ranges may be separated by new line or by comma. The end time must be greater than the start time. If no value is specified low priority messages are sent like normal SMS messages. E.g. 00:01 - 06:00, 20:00 - 23:59

Accounting records	Each time an SMS message has been transmitted or received an accounting record may be generated. Accounting records may be used for internal billing processes and give management a detailed view on the use of the server.
	The detailed outbound accounting records are further aggregated into aggregated accounting documents. These aggregated figures may be used (not yet implemented) in order to impose restrictions on users for sending further SMS messages.
	
	Values
	  ● Generate accounting records: this is the recommended value
	  ● No accounting records

Accounting database	Accounting documents - if enabled - are created in the configuration database or another database. Either way, the database must be based on the Sms4Domino template.
	Specify the filepath, relative to the Domino data directory, where the database resides. The database must be located on the same server as the configuration database.
	In this release accounting records must be created in the control database.
	
	Values
	  ● leaving the field empty implies that the configuration database is used for storing accounting records. This is the mandatory setting for the current release.
	  ● filepath (e.g. Sms4Domino\accounting.nsf) to specify a dedicated database

Proxy settings	Clickatell gateways are based on standard HTTP(S) . Direct HTTP(S)

	access to the Internet is not always available. In case a proxy server is used the proxy parameters are defined here. All Clickatell gateways share the same proxy settings. If no Clickatell gateway is used the settings are not relevant. Consult your network administrator for the proxy values.

Proxy type	Select a value from the list
	    Direct connection - no proxy is being used
	    HTTP(S) proxy
	    Socks V4 or V5 proxy

IP address	IP addressof the proxy server. This address must be reachable by Sms4Domino. Specify the address as 4 blocks of digits separated by dots (.): xxx.xxx.xxx.xxx (e.g. 192.168.2.105).
Port	Port number of the proxy server.
Username/password	Some proxy servers require authentication by a username and password. In Windows environments the Windows domain credentials my be used. In that case enter <domain>\<username> for the username.

SmsLib Settings

Sms4Domino runs on top of the open source SmsLib library (available at smslib.org). Parameters of SmsLib are documented on the web site smslib.org. For convenience reasons the documentation has been copied into this help document.

SmsLib parameters are normally defined via java properties files. In a Domino environment administrators prefer to define parameters in Notes fields on Notes documents. Sms4Domino reads the settings from the instance configuration document and translates them internally into SmsLib properties.

Parameter	Purpose
SERIAL_NOFLUSH	If true, SMSLib will not call the flush() system call upon writing to the serial port. This should normally be left as false, meaning that we need the flush() call. Set it to true for Linux/RxTx and virtual ports where the flush() call leads to errors Default: false
SERIAL_NOEVENTS	If true, SMSLib will not register the serial port callback events (i.e. framing errors, etc). This is reported to help with some Sony Ericsson models. Default: false
SERIAL_POLLING	If false, SMSLib will rely on the serial port interrupts to wake up and get incoming data. If true, if will create a separate thread in order to periodically poll the serial port for incoming data. Set it to true only on Linux/RxTx installations if you see that SMSLib behaves as if no modem is connected. Default: false
SERIAL_POLLING_INTERVAL	This is the polling interval. It has a meaning only when the SERIAL_POLLING is true, i.e. SMSLib works in polling mode. The value is in milliseconds. The default value is a good starting point - you may experiment with different values Default: 200ms
SERIAL_TIMEOUT	This is the master timeout set on the serial ports. The value is in milliseconds. Default: 30 000ms
SERIAL_KEEPALIVE_INTERVAL	This is the keep-alive interval. The keep-alive functionality in SMSLib periodically "pings" the modem in order to keep the connection active. The value is in seconds. Default: 60 s
SERIAL_BUFFER_SIZE	This is the buffer size SMSLib uses for the internal queues. The value represents characters (bytes). Default: 16 384 bytes
SERIAL_CLEAR_WAIT	This is a delay value, used when clearing the queues. The value is in milliseconds.

	Default: 1 000 ms
SERIAL_RTSCTS_OUT	If set, RTS/CTS will be enabled for outbound serial traffic as well. Useful for some types of phones
	Default: false
QUEUE_RETRIES	The retry count of the background queue manager for a message failing to be sent out. After this retry count, the queue manager will stop trying sending this message.
	Default: 3 times
AT_WAIT	This is a delay value, used after some AT commands. The value is in milliseconds.
	Default: 200 ms
AT_WAIT_AFTER_RESET	This is the delay value. SMSLib will sleep for this period after issuing a modem reset command, in order to give time to the modem to initialize. The value is in milliseconds.
	Default: 10 000 ms
AT_WAIT_CMD	This is a delay value, representing the time needed after issuing a "+++" command. The value is in milliseconds.
	Default: 1 100 ms
AT_WAIT_CGMS	This is a delay value, representing the time needed after issuing a "CMGS" command. The value is in milliseconds.
	Default: 200 ms
AT_WAIT_NETWORK	This is a delay value, representing the time that SMSLib will give modem in order to attach to the GSM network. The value is in milliseconds.
	Default: 5 000 ms
AT_WAIT_SIMPIN	This is a delay value, representing the time that SMSLib will give modem after the SIM PIN initialization. The value is in milliseconds.
	Default: 5 000 ms
CNMI_EMULATOR_INTERVAL	If CNMI fails, SMSLib will launch a separate thread for periodically polling the modem for inbound messages and still act as an asynchronous reader.
	Default: 30 s
OUTBOUND_RETRIES	This is the retry count, defining how many times SMSLib will try to send a failing message. After those retries, message will be marked as failed.
	Default: 3 times
OUTBOUND_RETRY_WAIT	The wait time between two attemps for sending a failed message. Value is in milliseconds.
	Default: 3 000 ms
WATCHDOG_INTERVAL	This is the process interval of the Service Watchdog background thread. The value is in seconds.
	Default: 15 s

MASK_IMSI	When true, the IMSI of the phone's/modem's SIM is not reported nor included in the logs (safety reasons). Default: true
CONCURRENT_GATEWAY_START	When true, SMSLib will try to start all defined gateways concurrently (to speed up start up time). When false, gateways start one after the other. Default: true
DISABLE_CMTI	When true, SMSLib will try to disable all unsolicited inbound notifications. Most modems do not implement the unsolicited notifications correctly. We recommend disabling them. Default: true
HOURS_TO_ORPHAN	This setting defines the age (in hours) after which SMSLib will treat a message part as "orphaned". Orphaned parts are parts of a big, multipart message which are received alone and their "sister" parts were never received in order for the full message to be reconstructed. Once a message part gets older than what the setting says, the relevant notification method is called. Default: 72 hours
DISABLE_CMMS	The CMMS command (Keep GSM Link Open) is used to increase the sending rate, but may cause instability on some modems. It is set by default. Default: false
DISABLE_COPS	The COPS command is used for network selection. The default should suffice, but you can disable the use of the command. Default: false
CACHE_DIRECTORY	The generic cache directory location used by SMSLib. Default: java.home
QUEUE_DIRECTORY	The cache directory used by the Queue Manager. If left undefined, the Queue Manager acts in a non-persistent way. If you set it to a path, Queue Manager will save pending outbound messages (i.e. queued messages) and reload them between SMSLib invocations. SMS4Domino uses its own caching (in Domino databases). There is no need for an additional queue directory Default: undefined

Gateway definitions

Sms4Domino uses SmsLib for sending and receiving SMS messages. Each individual communication channel (e.g. serial modem) is defined via a Gateway document in the configuration database.

Without at least one active Gateway document Sms4Domino will not be able to start.

Sms4Domino supports different gateway types (called Gateway Family): serial modems, IP modems, Internet SMS service providers, ... Gateways may be <u>outbound </u>- sending SMS messages from Domino to the rest of the world - or <u>inbound </u>- receiving SMS messages from anywhere in the world.

A gateway is linked to exactly one Sms4Domino instance.

The settings in the Gateway documents are read into memory when Sms4Domino starts. If you need to make changes to any of the settings you will have to quit and start the Sms4Domino application.

Procedure to create a Gateway document

- Open the Sms4Domino configuration database.
- Select *System -> Gateways* in the navigator. The right part of the screen displays all defined gateways.
    - Only Enabled gateways are used by Sms4Domino.
    - Gateway ID must be unique. All gateway related logging uses this id.
    - Used by: the Sms4Domino instance that uses the gateway
    - Family
    - The status of the enabled modems is updated in real time by Sms4Domino. This info is not valid when Sms4Domino is not running. You may need to refresh the view by clicking F9.
    - Status date: date/time when the status has been updated.
    - Description is not used by Sms4Domino
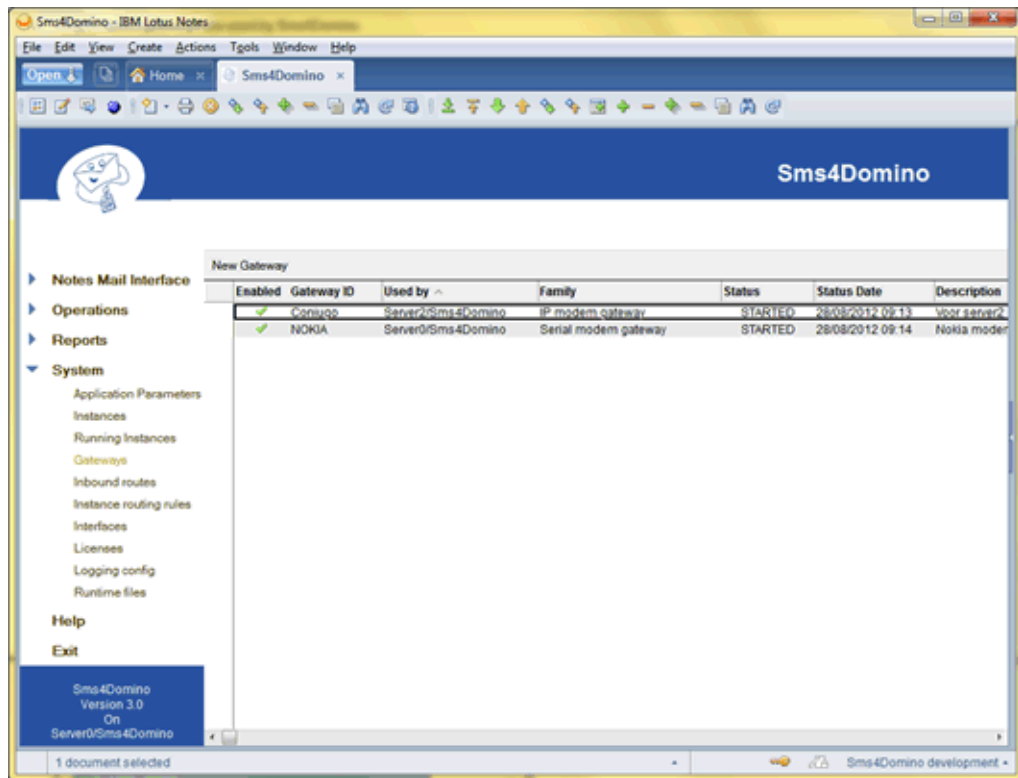
- Click on the action button *New Gateway* or double click on an existing Gateway document. Select the kind of gateway that you would like to create. This cannot be changed. Consult the release notes and verify what Gateways families are supported by the current release.
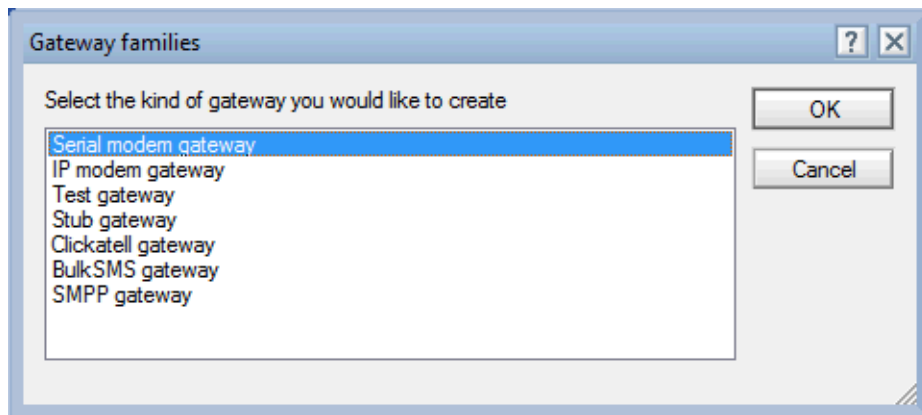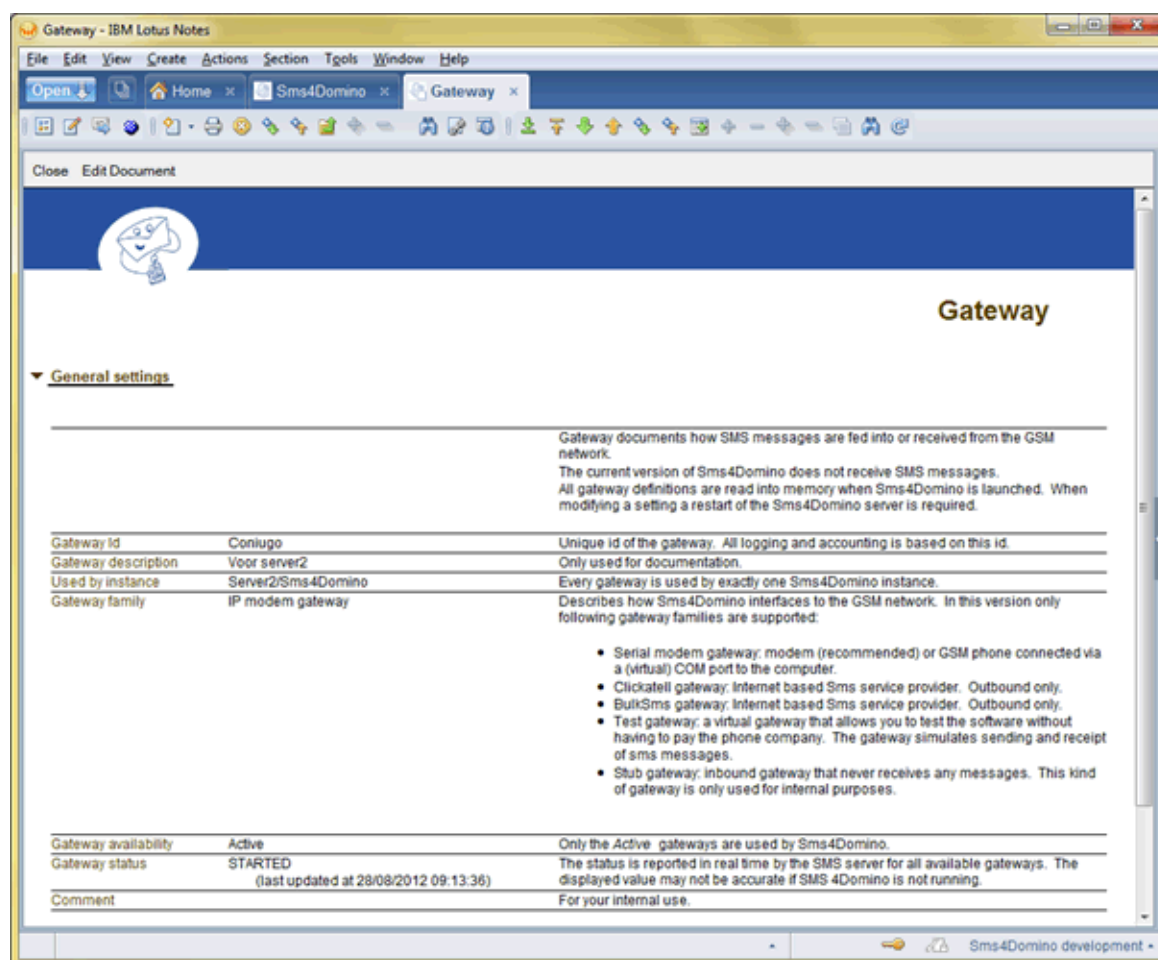
Gateway document settings

The settings are grouped in two sections: general settings that are required for all gateway families and specific settings that relate to the selected gateway family.

General settings

- Gateway id: Unique id of the gateway. All logging and accounting is based on this id.
- Gateway description: this free text field is only used by the administrator for documentation purposes.

- Used by instance: select the instance from the list of known instances. The selected instance will use the gateway.

- Gateway family: the selected gateway family. This value cannot be changed. The gateway family defines how the computer is connected to the GSM network.

  o Serial modem gateway: modem (recommended) or GSM phone connected via a (virtual) COM port to the computer. Read the help document on connecting a modem to the computer.

  o Test gateway: a virtual gateway that allows you to test the software without having to pay the phone company. The gateway simulates sending and receipt of sms messages.

  o Stub gateway: inbound gateway that never receives any messages. This kind of gateway is only used for internal purposes. There is no practical reason for you to create a stub gateway.

- Gateway availability: only *Active* gateways are used by Sms4Domino. In case of hardware problems you could disable a gateway. Remember to restart Sms4Domino after a parameter change.

- Gateway status: when Sms4Domino is started, the program will update the active gateway documents each time the status changes. The gateway status is managed by SmsLib and reported to Sms4Domino. Possible status values are

  o STARTING

  o STARTED

  o FAILURE

  o RESTART

  o STOPPING

  o STOPPED

- Comment: free text field

Gateway

▼ General settings

		Gateway documents how SMS messages are fed into or received from the GSM network. The current version of Sms4Domino does not receive SMS messages. All gateway definitions are read into memory when Sms4Domino is launched. When modifying a setting a restart of the Sms4Domino server is required.
Gateway Id	Coniugo	Unique id of the gateway. All logging and accounting is based on this id.
Gateway description	Voor server2	Only used for documentation.
Used by instance	Server2/Sms4Domino	Every gateway is used by exactly one Sms4Domino instance.
Gateway family	IP modem gateway	Describes how Sms4Domino interfaces to the GSM network. In this version only following gateway families are supported: • Serial modem gateway: modem (recommended) or GSM phone connected via a (virtual) COM port to the computer. • Clickatell gateway: Internet based Sms service provider. Outbound only. • BulkSms gateway: Internet based Sms service provider. Outbound only. • Test gateway: a virtual gateway that allows you to test the software without having to pay the phone company. The gateway simulates sending and receipt of sms messages. • Stub gateway: inbound gateway that never receives any messages. This kind of gateway is only used for internal purposes.
Gateway availability	Active	Only the *Active* gateways are used by Sms4Domino.
Gateway status	STARTED (last updated at 28/08/2012 09:13:36)	The status is reported in real time by the SMS server for all available gateways. The displayed value may not be accurate if SMS 4Domino is not running.
Comment		For your internal use.

Serial Modem Gateway documents

A Serial Modem Gateway document is required when a GSM modem is connected to the server via a serial port (COM port). The physical connection may be USB, Bluetooth or using a RS-232 cable. In production environments only the RS-232 connection should be used. The other connections are less stable and require additional configuration settings.

In this document we assume the modem is connected via a RS-232 cable to a serial COM port. The modem was tested and responded with OK to the AT command.

Specific settings for Serial Modem Gateway

- Directions: specify if the modem is used for sending SMS messages (outbound) and/or receiving SMS messages (inbound)

- COM port: select the COM port from the list or enter the name of the port. The name is case sensitive and should correspond to the name reported by Windows. Do not specify the colon after the COM port name

- COM speed: this speed must correspond to the speed that the modem is using. Some modems may operate at multiple speeds. It is recommended to test the communication to the modem at the specified speed via a test program (see the help document *Choice and installation of the modem*)

- Manufacturer - model: communication between the computer and the modem is based on AT-commands. Most modems are using the same set of AT commands. Some modems deviate from the standard. Specify *Generic* if your modem is not listed.

- PIN code: the PIN code of the SIM card in the modem. Sms4Domino may lock the card if a wrong Pin code is specified. Sms4Domino has no tools to reset the Pin code.

IP Modem Gateway documents

IP modems are serial modems that are connected via a serial-to-ethernet convertor. Some modems may have a build-in convertor, others require a separate box.

Sms4Domino sends and receives the same AT commands but uses TCP/IP to communicate with the modem.

Before configuring this kind of modem gateway document you should have tested the modem and configured the serial-to-ethernet convertor.

Specific settings for IP Modem Gateway

- Directions: specify if the modem is used for sending SMS messages (outbound) and/or receiving SMS messages (inbound)

- IP address: the IP address of the serial-to-ethernet convertor. It must be possible to ping this address from the machine where Sms4Domino will be running.

- IP port: the port number that has been configured in the serial-to-ethernet convertor. When issuing a telnet command (in binary mode) to the <ip address> <port> number you should be able to send AT commands and receive the responses.

- Manufacturer - model: communication between the computer and the modem is based on AT-commands. Most modems are using the same set of AT commands. Some modems deviate from the standard. Specify *Generic* if your modem is not listed.

- PIN code: the PIN code of the SIM card in the modem. Sms4Domino may lock the card if a wrong Pin code is specified. Sms4Domino has no tools to reset the Pin code.
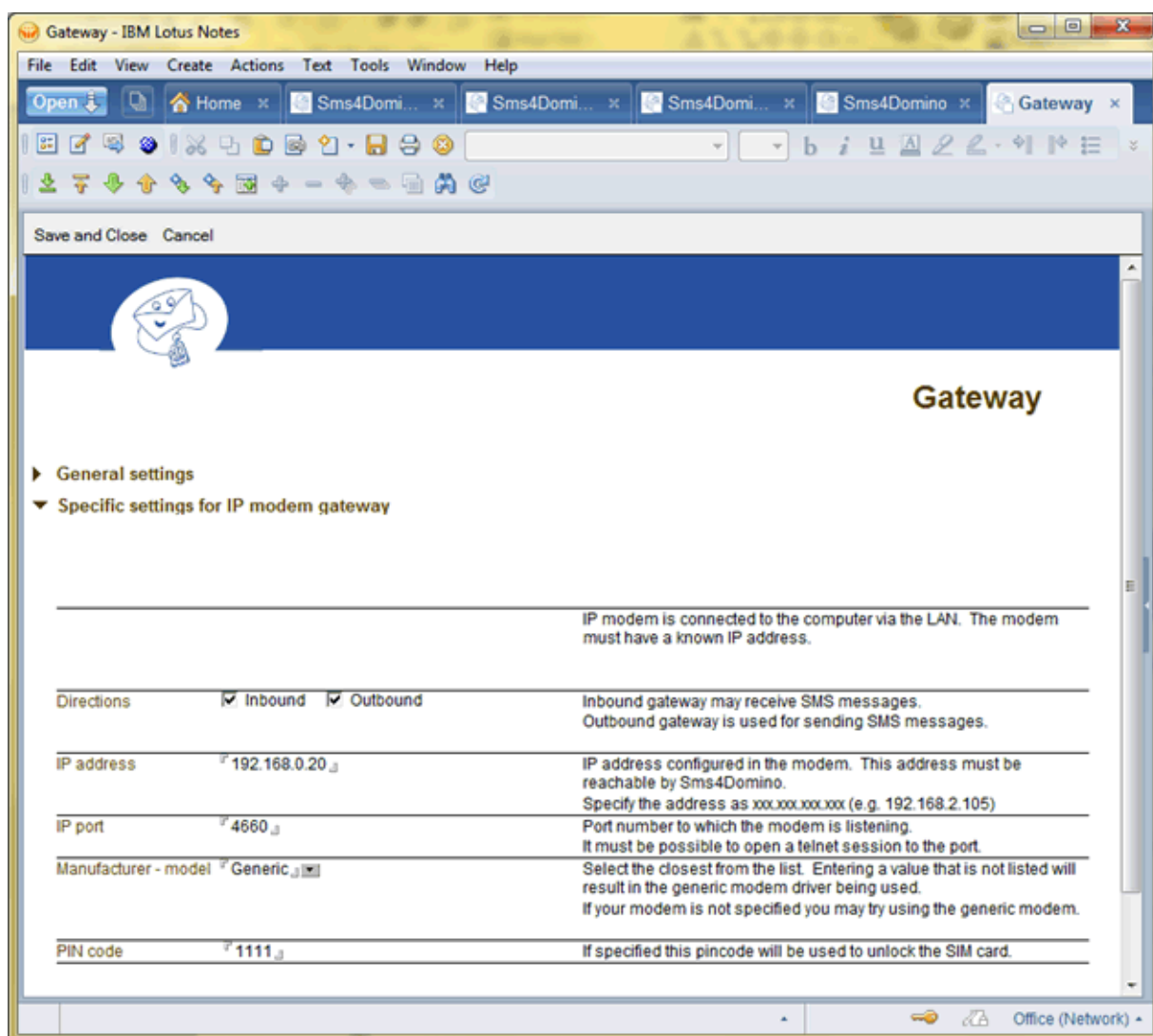
Clickatell Gateway documents

Clickatell is a SMS service provider. Sms4Domino uses the HTTP(S) protocol to send SMS messages via Clickatell. You must have a valid Clickatell account and sufficient credits.

The machine where the Sms4Domino program is being executed must have HTTP(S) access to the Internet. This connection may be direct or via a proxy server. Proxy server settings are configured in the application profile.

Specific settings for Clickatell Gateway

- Directions: the Clickatell Gateway is outbound only.
- User: the username for your Clickatell account.
- Password: the password for your Clickatell account.
- API Id: This is issued upon addition of an HTTP sub-product to your account. A single account may have multiple API IDs associated with it.
- Mode: select HTTP or HTTPS. Https is the preferred mode. Your username and password would otherwise be visible.
- Proxy settings: All communication with the Clickatell server goes via api.clickatell.com using the http or https protocol. This connection could either be direct or via a proxy. Proxy settings are shared between all http gateways and are defined in the application profile.

Test Gateway documents

A Test Gateway provides a low cost way for evaluating the software without incurring a cost from the phone company. Outbound test gateways pretend to send a Sms and may generate a transmission failure from time to time. Inbound test gateways pretend to receive a Sms message from time to time.

In production environments test gateways will not be used.

<u>Specific settings for Test Gateway</u>

- Directions: specify if the modem is used for sending SMS messages (outbound) and/or receiving SMS messages (inbound)

- Polling cycle: a thread polls an internal queue for new outbound SMS requests every x milliseconds.

- Receive cycle: only used fro inbound gateways. One virtual inbound SMS message is received per receive cycle.

- Fail cycle: If set to N then every Nth sent or received SMS message will throw an error or flag the outbound message as not sent. Set to zero to disable failures. Integer value.

- Outbound fail method: flag as unsent or throw an error.

- Outbound duration: Simulated duration for sending one SMS message. The actual duration will be a random time between the two specified values.
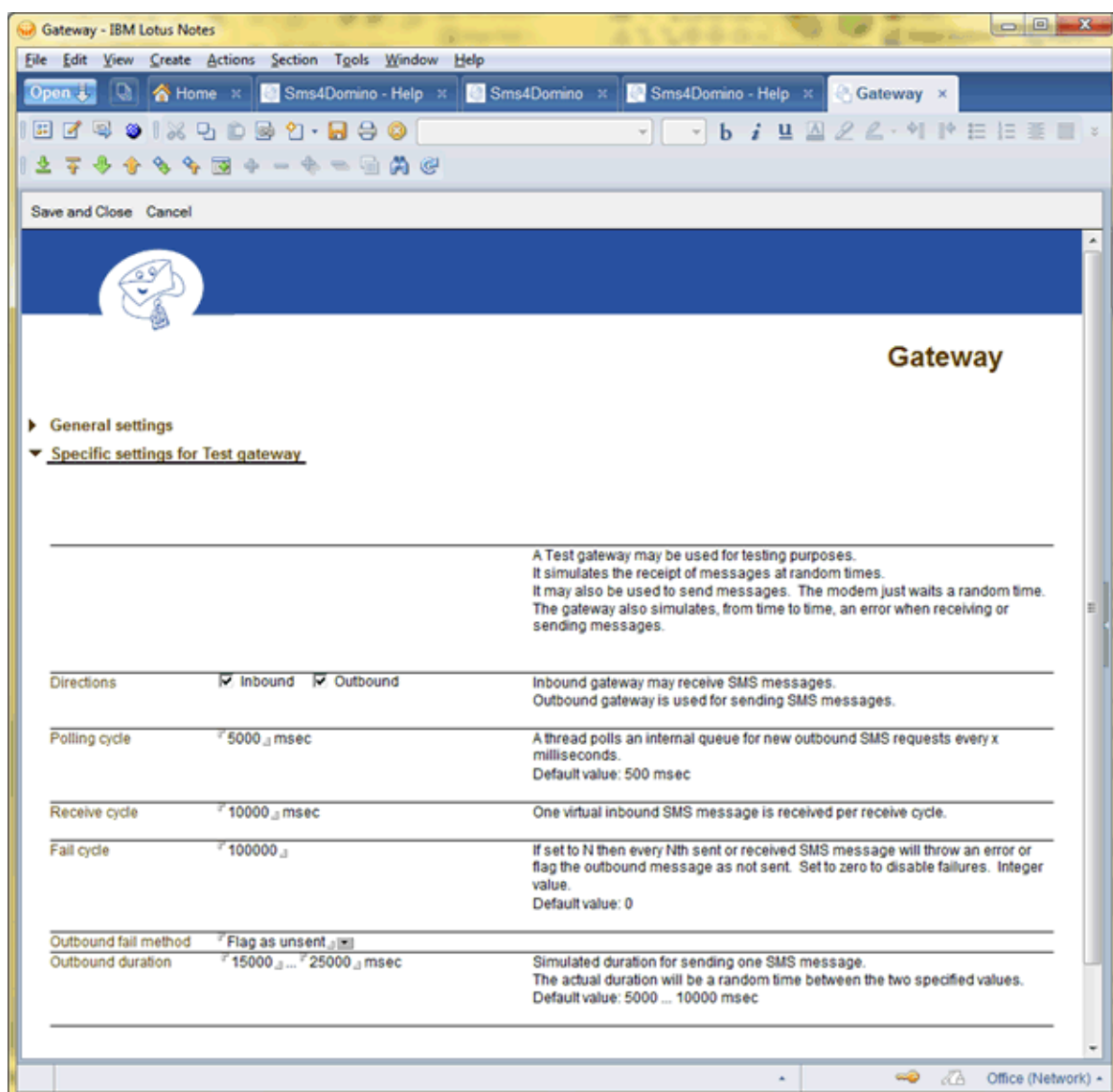
Stub Gateway documents

SmsLib, the communications package on which Sms4Domino relies, throws an error if no gateway has been defined. To avoid this unpleasant situation Sms4Domino uses a Stub gateway in case no other gateways are active.

A Stub gateway is inbound only and will never receive a message. It has no other practical use.

Specific settings for Stub Gateway

- Directions: stub gateways are always inbound and cannot be set to outbound
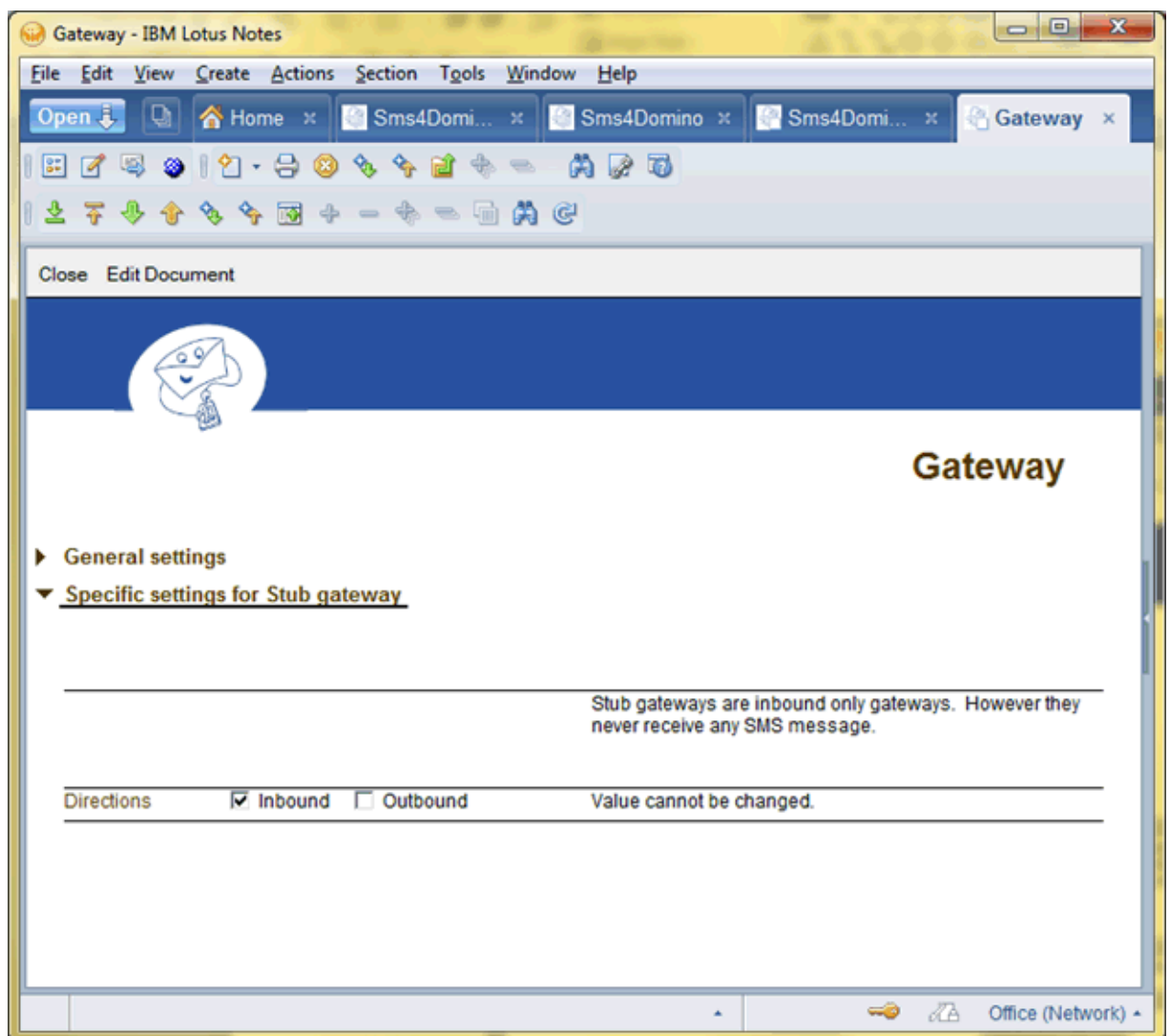
Interface definitions

All communications between Sms4Domino and the end users is done via Interfaces. There should be at least one active interface in the system.

Sms4Domino supports different interface types (called Interface Family): Notes mail message, HTTP, folder monitor, ... Interfaces may be <u>outbound</u> - sending SMS messages from Domino to the rest of the world - or <u>inbound</u> - receiving SMS messages from anywhere in the world. Outbound interfaces may or may not accept the returned delivery reports.

Interfaces have a priority. Sms4Domino will poll all interfaces in order of priority for Sms messages to be sent. As such, a high priority interface could block a lower priority interface from sending Sms messages. When two interfaces have the same priority the behaviour of Sms4Domino is not deterministic.

An interface is linked to exactly one Sms4Domino instance.

The settings in the Interface documents are read into memory when Sms4Domino starts. If you need to make changes to any of the settings you will have to quit and start the Sms4Domino application.

Procedure to create an Interface document

- Open the Sms4Domino configuration database.
- Select *System -> Interfaces* in the navigator. The right part of the screen displays all defined interfaces.
  - o Only Enabled interfaces are used by Sms4Domino. The enabled interfaces are listed first in descending order of priority.
  - o Interface ID must be unique. All Interface related logging uses this id
  - o Priority. A high numeric value means a high priority.
  - o Used by. The Instance that uses the interface.
  - o Family
  - o Flow: interface may be inbound or outbound. Outbound interfaces may accept or reject the inbound delivery reports.
  - o Description is not used by Sms4Domino
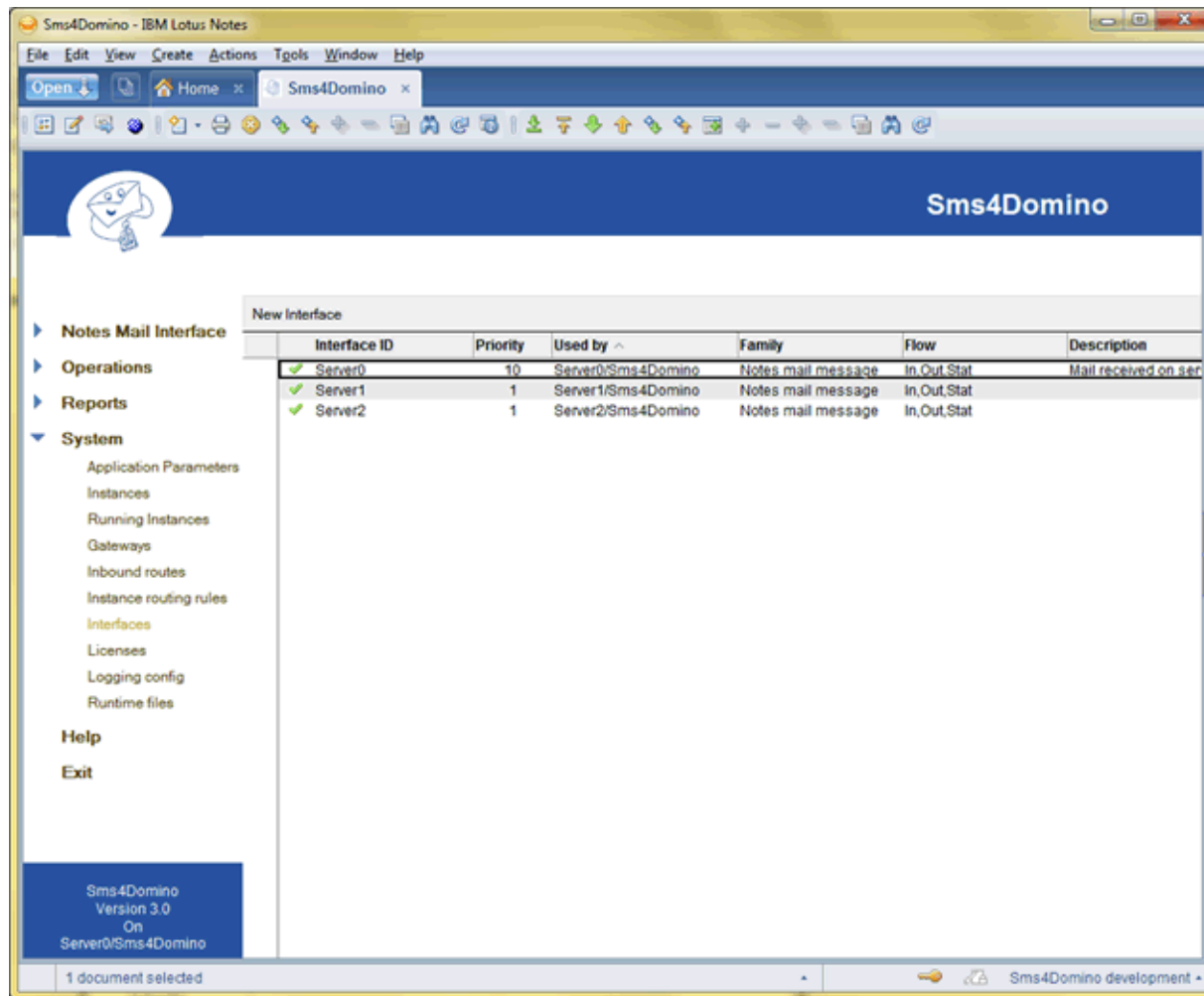
- Click on the action button *New Interface* or double click on an existing Interface document. Select the kind of interface that you would like to create. This cannot be changed. Consult the release notes and verify what Interface families are supported by the current release.
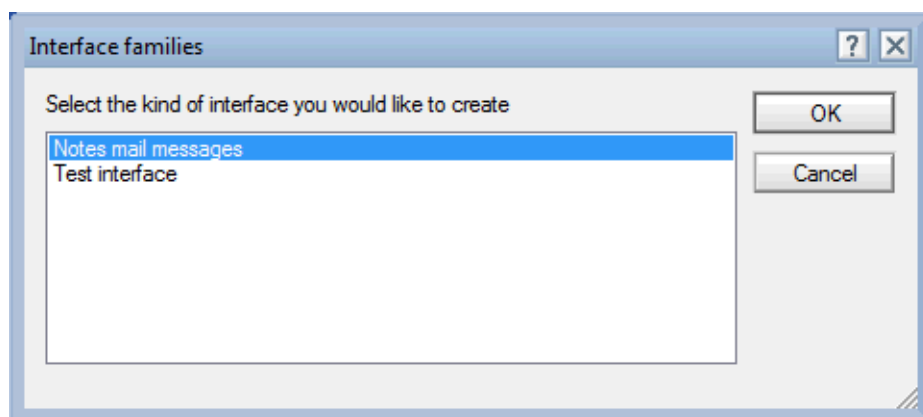
Interface document settings

The settings are grouped in two sections: general settings that are required for all interface families and specific settings that relate to the selected interface family.

General settings

- Interface id: Unique id of the interface. All logging and accounting is based on this id.

- Interface description: this free text field is only used by the administrator for documentation purposes.

- Used by instance: the Sms4Domino instance that uses the interface.

- Interface family: the selected interface family. This value cannot be changed. The interface family defines how end users or systems present their requests for sending Sms messages to the Sms4Domino application.

  o Notes mail messages: end users send an email to a foreign Domino mail domain (e.g. Name@873663@sms). The mail message is converted by Sms4Domino into an outbound SMS. Delivery reports are returned via mail to the sender of the mail

  o Test interface: an interface that never generates a request for sending Sms messages. This kind of interface has no practical purpose.

- Interface priority: a higher number means a higher priority. he highest priority interface is polled first for outbound SMS requests. Inbound SMS messages (received from the modem) are presented first to the highest priority interface. When two or more interfaces have the same priority they are polled in a non deterministic way.

- Interface flow: Specify what kind of messages may be generated by/delivered to the interface

  o Inbound: SMS message received from the GSM operator.

  o Outbound: SMS message generated by the interface.

  o Status Delivery Report: notifications received from the GSM operator as a feedback on a previous outbound message

- Interface availability: only *Active* interfaces are used by Sms4Domino. Remember to restart Sms4Domino after a parameter change.

- Comment: free text field

Notes Mail Interface documents

Notes Mail Interface documents provide a means to end user and systems to send Sms messages by simply sending an email. Sms4Domino extracts the text and destination from the mail and delivers it as one or - in case the text is larger than the max size of a single Sms - more Sms messages.

Requests for sending Sms messages are created by sending an e-mail to <name>@<gsm number>@<smsdomain> or <gsm number>@<smsdomain>. Domino mail routing routes the mail to the foreign domain <smsdomain>. A Notes database (e.g. sms.box) must be linked to the domain. The Notes Mail Interface reads the mails from the foreign domain mailbox and sends them out as Sms messages.

In order to use the Notes Mail Interface a foreign mail document must have been configured in the Domino directory. The procedure is explained in detail in the Domino help file. A short description is included in the Sms4Domino help file.

The Notes Mail Interface may also be used for processing Sms messages received from the mobile network (inbound Sms messages). Inbound Sms messages are converted into Domino mail messages and may be delivered by Notes mail to a user. Sms4Domino currently uses following algorithms to determine the destination for the Sms:

- Fixed routing: all Sms messages received from GSM number <gsm number> are route by mail to the mail address <recipient>
- Fixed routing based on message content: Sms4Domino reads the text of the received Sms. For example, you configure that "all messages containing **John** must be routed to Lotus Notes user John/Acme"
- Auto matching: this is best explained by an example. Assume John/Acme sends a Sms via Sms4Domino to Paula (Paula@0475321825@sms). When Sms4Domino receives a Sms from Paula's GSM within a given timeframe, Sms4Domino assumes that it is a reply for John and will send an e-mail to John/Acme.
  However if Paula is very popular and receives over a short timeframe Sms messages from John/Acme, Bill/Acme and David/Acme, Sms4Domino is not able to determine the destination for a received reply from Paula. In that case zero, one or more conflict handlers may be used. Following conflict handlers are available:
  - o Most recently used: suppose John has sent a Sms to Paula@0475321825@sms at 8:00 AM and Bill has sent one to the same number at 3:00PM. At 3:30 PM a return Sms is received from 0475321825. Sms4Domino routes the Sms to Bill's mailbox.
  - o Most frequently used: suppose John has sent 10 Sms messages to Paula@0475321825@sms during the last week. Bill has sent 1 Sms message to Paula@0475321825@sms during the same period. When a return Sms is received from 0475321825, Sms4Domino routes it to John's mailbox.
  - o Send to all: John, Bill and David have sent Sms messages to Paula@0475321825@sms during the last week. When a return Sms is received from 0475321825, Sms4Domino routes it to the mailboxes of John, Bill and David.
  - o Send to Sms4Domino administrator: Sms4Domino detects a conflict and sends the message (or just an alert) to the Sms4Domino administrator(s). The administrator may take manual action by forwarding the Sms.
- No forwarding: the Sms is kept in the system and should be forwarded manually. These Sms messages are displayed in the view *Operations -> Received - no match.*

Specific settings for Notes Mail message

- Database server & filepath: Name of the server - filepath of the sms.box.
  - o Leaving both fields empty corresponds to the control database being used as sms.box. This is the only acceptable setting for this release.
  - o Leaving the server name empty corresponds to sms.box on the same server as the

control database.

- o You cannot define more than 1 Notes Mail Interface for the same database (in this version of SMS4Domino there may only be one Notes Mail Interface)

- o The database must be based on the Sms4Domino template and not on the standard Domino mail router template. If needed, a custom made template could be used. However this template must contain a specific set of views (see next)

- o Sms4Domino program must have editor level access or higher to the database and the right to delete documents.

- New mail viewname: the Notes Mail interface scans at regular intervals this view for mails that must be transmitted as Sms. The view must have the same layout as the view *Mail* in the Sms4Domino template. The view is processed from top to bottom.

- Sms By Uuid viewname: Sms server tracks the sending of sms messages in response documents, linked to the received mail. The response messages are created in the sms.box database. Every outbound Sms message is identified by a UUID. This view lists all outbound mails sorted by UUID. The design of the view must be based on the *vwSMS_ByUuid* view as defined in the Sms4Domino template.

- Sms By Refnr PhoneNr viewname: Message delivery confirmations and failures must be matched to a past outbound Sms message. The match is based on the Reference Number received from the SMSC, concatenated with the recipient's phone number. The design of the view must be based on the *vwSMS_ByRefnrPhonenr* view as defined in the Sms4Domino template.

- Polling interval: the Domino router deposits mail messages in the sms.box of the Sms server. The requests are converted into another format by a separate thread. Specify the polling interval (in seconds) for this thread.

- Process mails deposited by router on server(s): In most cases this field should contain the name of the Domino server where Sms4Domino is running.
  The Domino router may deposit mails in Sms.box of any server (depending on the foreign domain documents). All Sms.box databases replicate within the Domino domain. Specify what mails should be processed by the current mail interface instance. In most cases there is one Sms4Domino instance per Domino server that contains a replica copy of Sms.box. Each instance should then process the mails deposited on the instance server. In a multi Sms4Domino server configuration there should exactly be one mail interface for each server that contains a replica copy of the Sms.box. This setting is also used to determine what mails should be removed from the database (see parameter Sms outbound retention time)

- Mail filter settings: the Domino mail router may send different kind of mail messages. The most common mails are based on the form *Memo* or *Reply.* But also meetings invitations, failure reports, message recall, ... may be deposited by the router in any mailbox. The mail filter settings section makes it possible to filter out these unwanted mails. The settings are optional. Both a whitelist and/or blacklist may be specified. The filter is defined either as a @formula or as a role. The @formula will be executed on the received mail message. The result of the formula must be either @true or @false. The sender of the mail must have the specified role name enabled in order to send Sms messages via this interface. Role names are written between [and].
  When a mail is received the whitelist formula and role check are executed first. In case of execution error or if the formula does not return @true the mail is rejected. In case the mail passed the whitelist formula the blacklist formula and role check - if present - are executed. In case of execution error or if the formula returns @true the mail is rejected. Rejected mails may either be silently discarded (although some logging will occur) or a non delivery failure is returned to the sender.
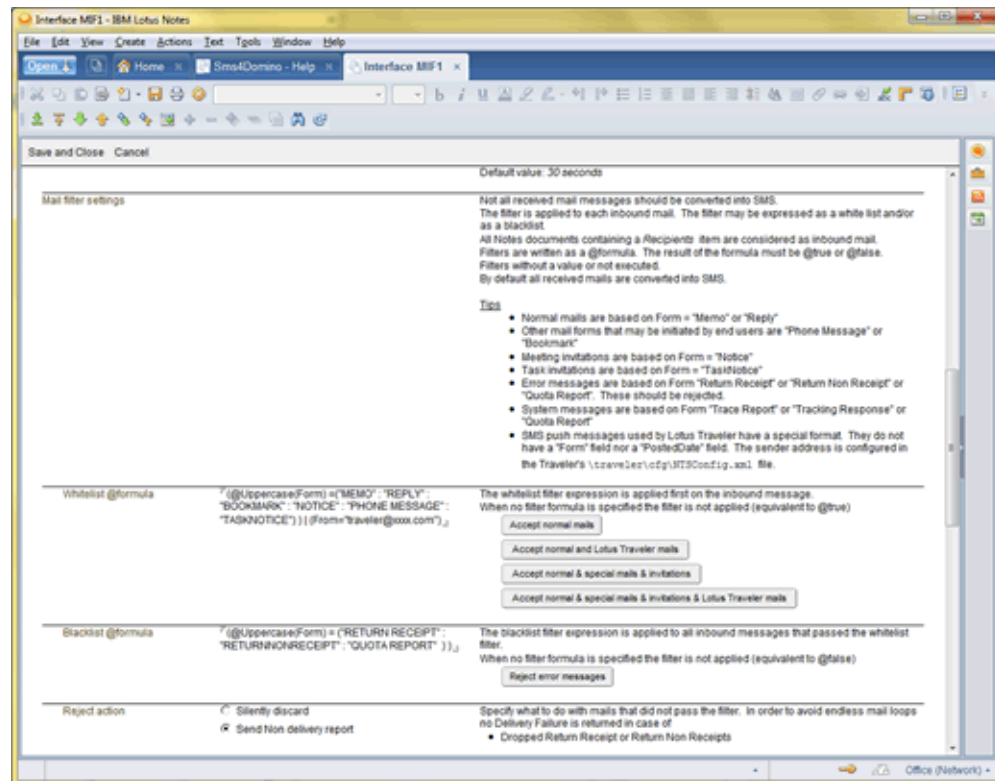
<u>Special precautions for implementing Lotus Traveler® SMS push</u>: Sms4Domino may be used for implementing IBM® Lotus Notes® Traveler Auto Sync via Sms (Short Message Service) push. This setting may greatly increase a device's battery life. By default, Lotus Notes Traveler® maintains a TCP connection between a device and server that allows both to stay in sync with each other. Depending on several factors (such as network provider, NAT timeouts, and device characteristics) maintaining this TCP connection can burden the device's battery, whereas using Sms mode for Auto Sync may decrease the burden on the device's battery and increase the amount of time between battery recharges.

Lotus Notes Traveler® sends mail messages each time a Sms must be forwarded to the smartphone. During testing we discovered that the mail does not contain a *Form* or *PostedDate* item. Some special whitelist formulas have been provided to accept the badly formed mail messages generated by Traveler®.

- Extract Sms messages from: Sms messages are text only. The contents will be extracted from either the subject, the unformatted body of the mail or the subject followed by the unformatted body of the mail. It is also possible to specify a @formula. The formula is executed in the context of the received mail. Some default formulas may be inserted by clicking an action.

- Truncate to: The resulting message is truncated to the specified size. If <= 0 then no truncation occurs. It is strongly recommende to specify some safety value. Large messages may result in a multi-message Sms.

- Outbound Sms retention time: when a delivery report (failure or confirmation) is received from the SMSC, it must be matched with an outbound Sms. Outbound Sms messages are deleted automatically. Delivery reports received after this deletion will not be delivered to the end user. The retention time parameters are expressed as positive integer multiples of the polling interval.

   o   Test every n polling intervals: the test for outbound messages that must be deleted is performed every n polling intervals. A value too high could result in too many old messages in sms.box. A value too low could overload the server.

   o   No delivery report requested: even when no delivery reports is requested it could happen that a negative delivery report is received. Negative value implies no

automatic deletion. Zero value implies automatic deletion after sending the Sms.

    o   Delivery report requested: it could take some hours or days before the delivery report is received. Negative value implies no automatic deletion. Zero value implies automatic deletion after sending the Sms.

- Pseudo sender (delivery report): mail address used as sender when Sms4Domino returns an email related to a delivery report) to a Notes user

- Pseudo sender (inbound Sms): Sms4Domino encourages active Sms conversations between the Lotus Notes user and the GSM owner. The Notes user must be able to send a reply on a received Sms. When Sms4Domino converts an inbound Sms received from 0475321825 to a Notes mail message, the sender field is constructed as Sms4Domino@0475321825@Sms. The first and last part of this three part mail address is configure here. Note that the third part must correspond to the name used in a foreign domain document in the Domino directory.

Phone number preprocessing

Introduction

End users may specify the phone number in different ways. The number 0474 96 11 11 in Belgium (country code 32) may be written as

+32474961111

0032/(474) 96 11 11

0474-96.11.11

For the correct matching of future inbound SMS messages with the outbound SMS it is essential to convert SMS numbers to a standard representation. Sms4Domino uses the international number format as a standard representation. This is composed of the plus (+) sign, the country code (1, 2 or 3 digits) followed by the local number.

End users may specify a badly formatted number (e.g. containing letters or too short). In case of obvious errors the SMS request should not be routed to the modem but must be returned immediately as a non delivery failure.

The administrator may impose restrictions on the allowed destinations. In the current release these restrictions are imposed at the level of the interface and will be applied to all users.

Details

The preprocessing of phone numbers is executed in a number of consecutive steps:

- Standardisation of the phone number.

    - Automatic convertion to the international number format. Optional reject of mall-formed numbers. The automatic convertion is based on the open source code of Google. It is recommended to activate this processing step. In most cases this is the only required standardisation step.

    - Ad-hoc convertion using a set of regular expressions.

- Validation of the result. The resulting number should match a regular expression

- Imposing restrictions on the resulting number. This is implemented using whitelist or blacklist regular expressions
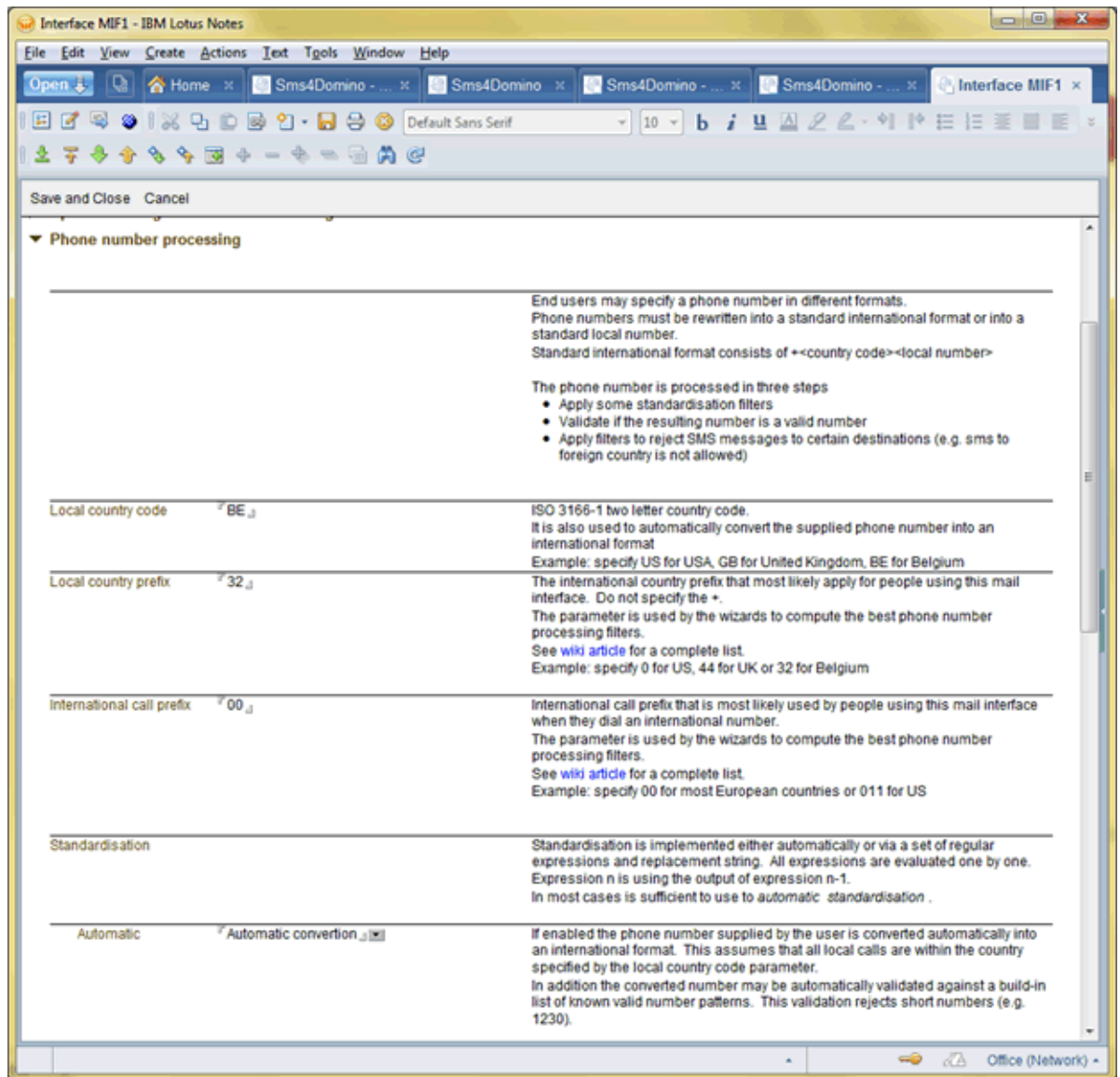
We will now explain all the available settings in the *Phone number processing* section of the mail interface document.

- Local country code: the automatic convertion logic needs to know the local two-letter country code. When the user specifies a local number (e.g. 0474961111) it will be converted into +<country prefix><local number> (e.g. +32474961111). The list of country codes is maintained by www.iso.org.
- Local country prefix: the pre-configured actions for manual convertion depend on the knowledge of the country prefix (1, 2 or 3 digits). The list of prefixes is documented in a Wiki Article (
  http://en.wikipedia.org/wiki/List_of_mobile_phone_number_series_by_country)
- International call prefix: some end-users specify an international number as <international call prefix><country prefix><local number> (e.g. 0032474961111). Some of the pre-configured actions for manual convertion are using this field in order to convert this prefix to the + sign.
- Standardisation
  - The automatic standardisation (based on Google's code) may be enabled or disabled. The logic may be applied in any of the following flavors
    - Do not perform automatic convertion
    - Convert the number automatically
    - Convert the number automatically and reject numbers that are most likely invalid. For example Google may know that phone numbers in Belgium do not start with

0411. So it will reject the local number 0411961111. However we cannot guarantee that Google's knowledge will remain up to date. This option could prove to be too restrictive.
- Manual convertion may be performed as a next step. It is configured as a set of filters (0 to 10 filters). Each filter will act on the result of the previous one

Standardisation	
Automatic	⌜Automatic convertion - reject invalid numbers ⌟ ▼
# ad hoc filters	⌜1 ⌟ ▼
Matching pattern 1	⌜[^0-9+] ⌟
Replacement pattern 1	⌜ ⌟
Replacement options 1	⌜Replace All Matches ⌟ ▼
Comment	⌜Remove all from the phone number that is not a digit or a + sign ⌟

- # ad hoc filters: specify the number of filters that will be used
- Each filter is based on Java regular expressions. The matching pattern tests if the phone number contains a specified pattern. If so the matched substring is replaved by the replacement pattern. Replacement options specify if only the first match or all matches must be replaced. The comment field is not used by Sms4Domino.
Working with Java regular expressions require some Java knowledge. The interface document contains some commonly used filters

Quick filters
Set filter number ⌜1 ⌟ ▼

Remove non digits

Remove space, point, comma ...

Replace international prefix by +

- Validation: when all standardisation steps have been applied the resulting number should be a valid international number. The validation regular expression specifies the format for a valid number.

Validation		
Validation patterns	^(\+)?\d{10,}$	Zero or one regular expressions. The phone number should match the pattern. If not the end user will receive an error message by mail. If none specified all numbers are considered valid. More validation will be done by the modem and/or the GSM network [Optional international prefix - at least 10 digits]
Comment	Only the optional international prefix followed by sequence of digits is allowed.	

- Destination restrictions: in this section you specify regular expressions for the patterns that are allowed (e.g. only sms to the local country -> ^\+32.*) or that are rejected. More than one pattern may be specified. In case no pattern is specified the test is not performed. The regular expressions are easy to write as we know that the phone number is now in the international format.

Destinations restrictions		
Allowed destinations	^\+32.* ^0032.*	Zero, one or more regular expressions. Every expression is entered on a separate line. If none specified all destinations that are not blocked are allowed. [Only local calls]
Comment	Only local calls are allowed. Only local calls are allowed.	
Rejected destinations	^(\+)?\d{12,}$ ^(\+)?\d{0,9}$	Zero, one or more regular expressions. Every expression is entered on a separate line. If none specified all destinations matched by the allowed destinations pattern are allowed. [Reject numbers > 12 digits] [Reject numbers < 10 digits]
Comment	Reject numbers > 12 digits. Reject numbers < 10 digits.	

Inbound SMS processing

Sms4Domino presents all received Sms messages to every interface linked to the Instance. Each interface decides if the Sms can be received. If none of the interfaces is capable of handling the inbound Sms, the Sms is processed according to the settings in the *Instance configuration* document. The Sms may for example be stored in the configuration database (view *Operations -> Received - no match*)

Sms4Domino currently uses following algorithms to determine the destination for the Sms:

- <u>Fixed routing</u>: all Sms messages from a specified phone number should be routed to a fixed mail address. This mail address can be any valid Lotus Domino® address. The fixed routing documents are visible in the view *System -> Inbound routes*.

- <u>Fixed routing based on message content</u>: Sms4Domino reads the text of the received Sms. For example, you configure that "all messages containing **John** must be routed to Lotus Notes user John/Acme".

- <u>Auto matching routing</u>: this automatic routing is based on the assumption that, when person x sends an Sms to person y it is very likely that a reply from y should be routed to person x. For each outbound Sms message, Sms4Domino creates a auto matching route. This document specifies that Notes user x has sent a Sms to number y. When a inbound Sms is received all routing documents are consulted based on the sender's phone number. Three outcomes are possible

  - No match: the inbound Sms cannot be delivered

  - Single match: the inbound Sms is routed via Domino mail to user x

  - Multiple matches for different users: there is a conflict. Zero, one or more conflict handlers may be invoked. Following conflict handlers are available:
    - o <u>Most recently used</u>: suppose John has sent a Sms to Paula@0475321825@sms at 8:00 AM and Bill has sent one to the same number at 3:00PM. At 3:30 PM a return Sms is received from 0475321825. Sms4Domino routes the Sms to Bill's mailbox.
    - o <u>Most frequently used</u>: suppose John has sent 10 SMS messages to Paula@0475321825@sms during the last week. Bill has sent 1 SMS message to Paula@0475321825@sms during the same period. When a return Sms is received from 0475321825, Sms4Domino routes it to John's mailbox.
    - o <u>Send to all</u>: John, Bill and David have sent SMS messages to Paula@0475321825@sms during the last week. When a return Sms is received from 0475321825, Sms4Domino routes it to the mailboxes of John, Bill and David.
    - o <u>Send to Sms4Domino administrator</u>: Sms4Domino detects a conflict and sends the message (or just an alert) to the Sms4Domino administrator(s). The administrator may take manual action by forwarding the SMS.
- <u>No forwarding</u>: the Sms is kept in the system and should be forwarded manually. These Sms messages are displayed in the view *Operations -> Received - no match.*

Auto matching routing documents are created automatically by Sms4Domino each time an outbound Sms is sent. Some of the settings of these documents are defined in the *Interface* document.

Auto matching routing documents may be removed automatically after a specified number of hours. A different period may be set for auto matching documents that are part of a routing conflict. This action is performed by the main polling thread of the mail interface.

All settings are documented in the *Interface* document.

Fixed routing	Active	Fixed routing documents are only consulted if the feature has been enabled.
Auto matching	Active	Sms4Domino creates an Auto-matching Inbound Routing document each time a SMS is sent. These documents are only created and consulted if auto matching feature has been enabled.
Document settings		Settings for the created Auto-matching Inbound Routing documents
Weight	1	Weight setting in the created Inbound Routing document.
Bind to gateway	Valid for all gateways	Should the gateway be part of the criteria in the created Inbound Routing document
Processing settings		Following parameters are used when performing the auto-matching algorithm
Validity period	1.440 minutes	When John sends a SMS to Jane then Jane must reply within this timeframe. Later replies cannot be automatched. A value <= 0 considers that all inbound routing documents are valid. 1 hour = 60 minutes 1 day = 1 440 minutes Default value: -1 minute
Conflict handling algorithms Frequent (Match on most frequent)		A conflict may be detected when an inbound SMS is received. Sms4Domino may try to make a best guess in solving the conflict. Zero, one or more algorithms may be activated. Specify the set of algorithms to be executed and the sequence of their execution. All algorithms will be executed in the specified sequence until an acceptable guess is found. If none of the algorithms results in a result the inbound SMS will be processed by a lower weight inbound routing document and finally according to the settings in the Sms4Domin instance configuration document. Available algorithms: • Recent (Match on most recent)

We will now explain all the available settings in the *Inbound SMS processing* section of the mail interface document.

- Fixed routing: if disabled Sms4Domino does not look at the fixed routing documents when an inbound Sms has been received.
- Automatching: Sms4Domino creates an Auto-matching Inbound Routing document each time a SMS is sent. These documents are only created and consulted if auto matching feature has been enabled.
- Weight: the weight setting in the created *Auto Matching Inbound Routing* document. The use of *weight* is discussed in the *Inbound routing* section of this help document.
- Bind to gateway: should the gateway be part of the criteria in the created Inbound Routing document. The use of *gateway* is discussed in the *Inbound routing* section of this help document.
- Validity period: automatching documents older than the validity period are not considered when an inbound Sms is received.
- Conflict handling algorithms: A conflict may be detected when an inbound SMS is received. Sms4Domino may try to make a best guess in solving the conflict. Zero, one or more algorithms may be activated. Specify the set of algorithms to be executed and the sequence of their execution. All algorithms will be executed in the specified sequence until an acceptable guess is found. If none of the algorithms results in a result the inbound SMS will be processed by a lower weight inbound routing document and finally according to the settings in the Sms4Domino instance configuration document. Available algorithms:
    - o   Recent (Match on most recent)
    - o   Frequent (Match on most frequent)
    - o   All (Match on all)
- Additional settings used when the most recent conflict handler is invoked
    - o   Sms4Domino looks into all conflicting Auto matching routing documents for the document that was most recently updated (date stored in the field Last outbound SMS in the Interface definition document). However the Last outbound SMS date must be at least Minimum time delta minutes later than the Last outbound SMS date of all other conflicting Auto matching routing documents.
    - o   Warning message: text that gets appended to the mail. It may be usefull to append "Warning: it was not clear who should get this Sms. If the Sms is not for you please try to forward it to a more appropriate person."
- Additional settings used when the most frequent conflict handler is invoked
    - o   Sms4Domino looks into all conflicting Auto matching routing documents for the document that was most frequently used (value stored in the field Outbound SMS count in the Interface definition document). However the Outbound SMS count must

be at least Minimum count delta % percent higher than the Outbound SMS count in all other conflicting Auto matching routing documents.

- o Warning message: text that gets appended to the mail. It may be usefull to append "Warning: it was not clear who should get this Sms. If the Sms is not for you please try to forward it to a more appropriate person."
- Additional settings used when the match on all conflict handler is invoked
  - o The inbound SMS will be mailed to all users (or to a subset) specified in all conflicting Automatching route documents. The set of conflicting Automatching route documents is reduced to this number.
- Automatic removal of auto matching routing documents. It is possible to specify a different period for valid versus conflicting auto matching routing documents.

Optional agent preMailDelivery

The Sms4Domino database contains a pre-mail-delivery agent called preMailDelivery. It tries to correct received mails that do not follow the Domino conventions:

- In case the mail does not contain an item called *Form* the item is added to the mail with value *Memo*

- In case the mail does not contain an item called *PostedDate* the item is added to the mail with value *@Now*

By enabling the agent you might simplify the configuration of the *Mail Filter Settings* in the Mail interface configuration document
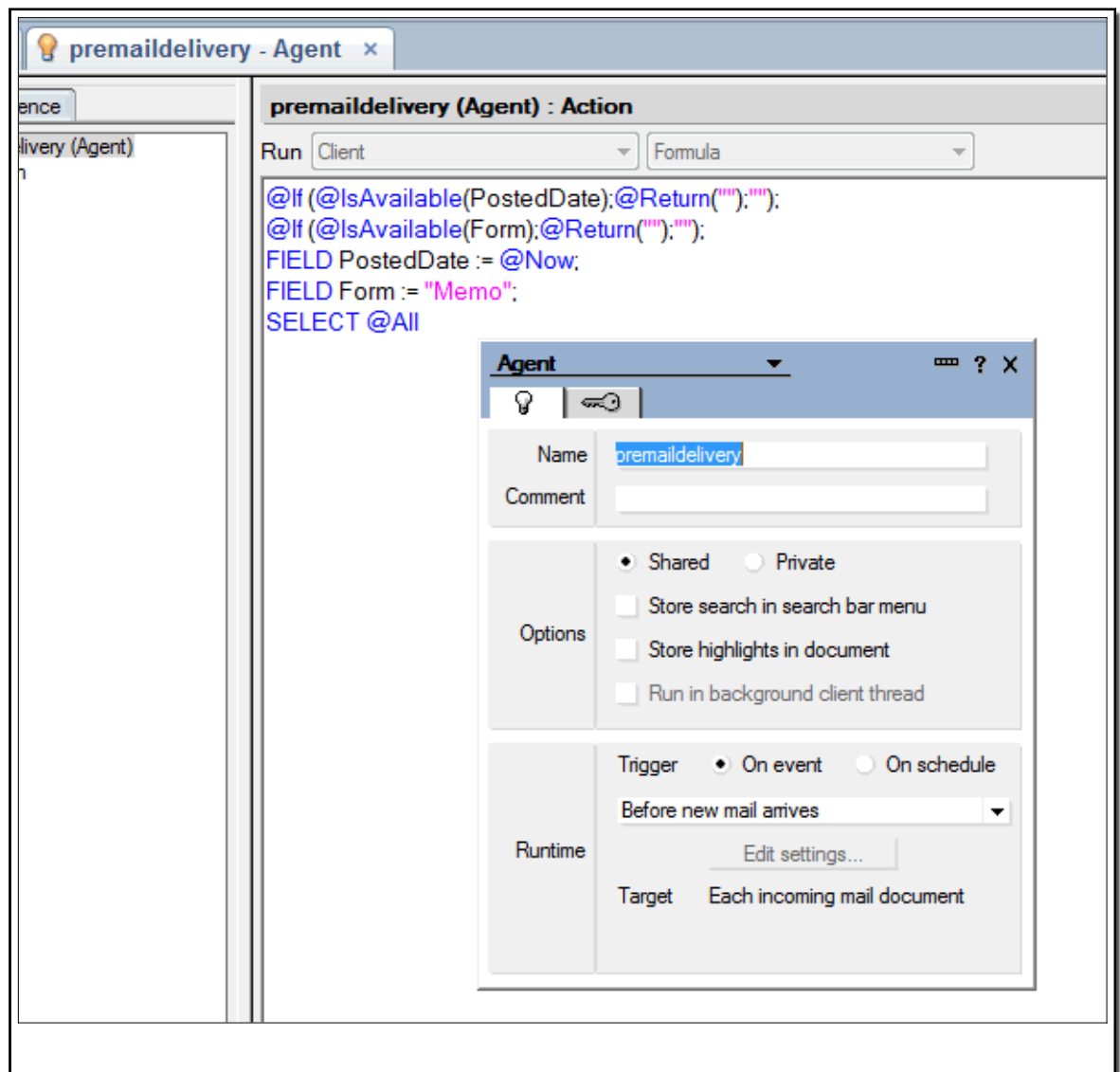
Inbound Route definitions

Inbound route documents describe how a Sms received from the GSM network should be processed. In this release inbound Sms messages may only be delivered via email to a Lotus Notes® user.

In case automatching has been enabled in the mail interface document, Sms4Domino creates automatically Inbound Route documents. These may be complemented by fixed route documents. Fixed route documents have precedence over the auto matching route documents.

Inbound route documents are read from the configuration database when needed. You may create/update route documents without the need to restart Sms4Domino.

How are Inbound Route documents used by Sms 4Domino

When an inbound Sms is received from the GSM network Sms4Domino looks for the presence of *Inbound Route* documents. The documents are ordered by *weight*: the heighest weight documents are consulted first.

Routing documents are skipped if the type of routing document has been disabled in the definition of the *Interface*. Routing documents are skipped if they do not match the received Sms (either based on the sender's phone number, the content of the Sms or the gateway that received the Sms).

There could be zero, one or more matching heighest weight routing documents.

- Zero documents: the process is repeated for the second heighest weight documents. If there are no more *Inbound Route* documents the received Sms is processed according to the settings in the *Instance Configuration* document.

- Exactly one document: this document determines how the inbound Sms will be transformed into a mail message.

- More than one document: this is a conflict situation. There are no conflict handlers for *Fixed routing* documents. Conflicting *Automatching routing* documents may be resolved by one or more conflict handlers. Either the conflict gets resolved or the process gets repeated for the second heighest weight documents.

Example

Company Acme, active in recruitment, has just given a GSM to the secretary of the CEO and CFO. When a Sms is received from one of the secretaries by Sms4Domino it should be routed to the CEO or CFO. Fixed routing must be configured.

Acme also launched a contest. People may send a Sms with a reply to a question that is published on the web site. The Sms must start with **CONTEST**. The secretary of the CEO is also allowed to participate in the contest. Content based fixed routing must be configured.

Recruiters have lists of GSM numbers of potential candidates. Each recruiter uses Sms4Domino to send daily about 100 Sms messages to potential candidates. When the candidate replies to such a Sms, the return Sms should be routed to the recruiter. The recruiters are also allowed to participate in the content. Auto-matching routing must be configured.

It is important to assign the heighest weight to the content based fixed routing document. Then there are two fixed routing documents for the secretaries. Standard auto-matching is used for the recruiters.

Procedure to create a Inbound route document

- Open the Sms4Domino configuration database.
- Select *System -> Inbound routes* in the navigator. The right part of the screen displays all defined routes.

o Only Enabled routes are used by Sms4Domino.

o Interface ID. The routing document will only be used by the specified id.

o Weight. The routing documents are processed in descending weight order.

o Type: auto matching route of fixed route

o From - route to: a route specifies that all SMS messages received from a specified phone number should be mailed to the specified Notes user. Phone numbers are expressed as regular expressins.

o Created on: date/time when the route document has been created. This date is used in order to automatically delete the expired Auto Matching Route documents.
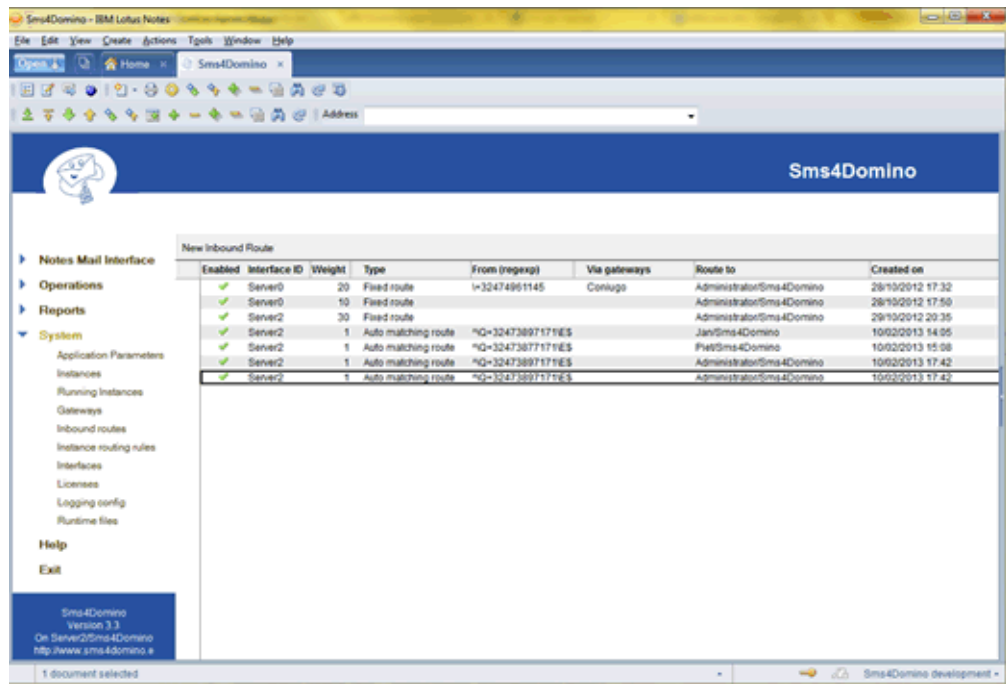
● Click on the action button *New Inbound Route* or double click on an existing Route document.

Inbound Route document settings

Inbound route documents are only processed by the Mail Interfaces.

● Type: Auto Matching route documents are created and deleted by Sms4Domino. As an administrator you create Fixed Route documents.

    o *Fixed route* documents are used first when fixed inbound routing has been enabled. The route documents must be created manually. There may only be one route that links a *from* number to a *destination*. There are no conflict handlers for fixed route documents.

    o *Auto matching* route documents are created automatically each time a outbound SMS message has been sent via the mail interface. The documents are removed automatically after a configurable time.

● Rule weight: routing documents are evaluated one set at a time. First the set of documents that have the heighest weight.

● From: Phone number pattern of the received SMS message. Must be specified as a

regular expression. The number is matched against the phone number of the SMS. This number is formatted as a international number without punctuation. (e.g. +32474957714). Leave the field empty to suppress the test on the phone number.

- Received via gateway: Inbound SMS messages enter the Sms4Domino system via a gateway (typically a GSM modem). Specify all gateways that match the current route. Leave the field empty to suppress testing the gateway.
- Route availability: only *Active* inbound route documents are used by Sms4Domino.
- Processing options: When a received Sms matches the settings of a routing document, the processing options of the routing document determine what should be done with the received Sms. In case of a routing conflict, a conflict handler may retain multiple matching routing documents (e.g. match all conflict handler). In that case all processing options are grouped together.

  o Route to destination: Mail address of the destination. Any valid Lotus Notes® address may be specified (e.g. John/Acme or john@acme.com or sales). Forwarding to a mobile phone is possible by specifying a valid Sms 4Domino mail address (e.g. John@0474961178@sms). Make sure not to create a loop.

  o Send alert to: Alert will be mailed to this address. Specify 0 or 1 valid mail addresses. The alert does not contain the text of the Sms.

  o Sms retention: The Sms may be kept in the Sms4Domino database. This allows the administrator to manually forward the Sms.

- Created by - created on: who created this document - when has it been created.
- For interface: a route document applies to a single Mail Interface. In larger configurations, having multiple interfaces, it may be necessary to duplicate a given fixed route document.
- Comment: free text field
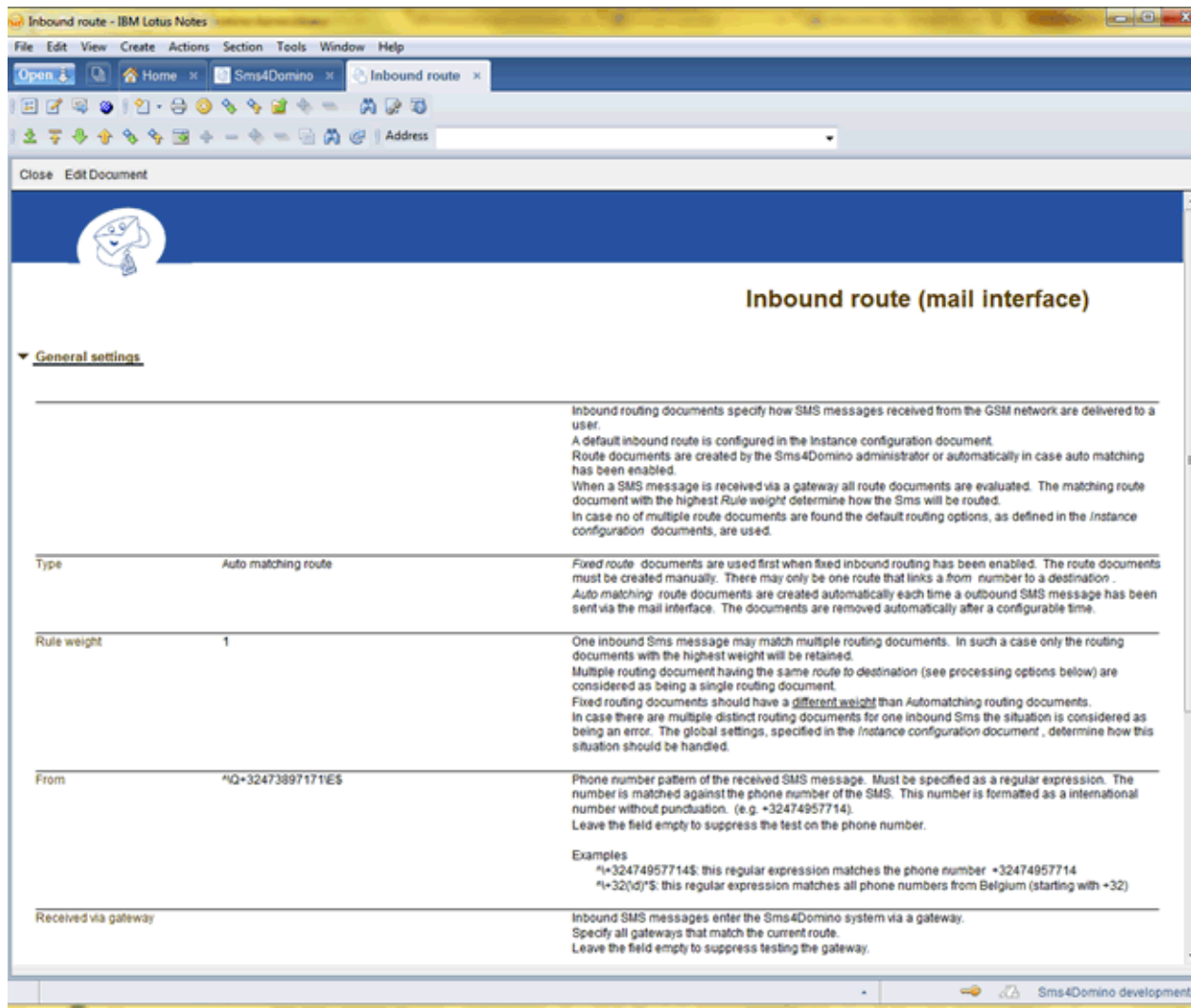
Specific settings for *Auto Matching route* documents

The conflicts handlers must know when the route document was last used and how many times it has been used.

- Last outbound Sms: when a outbound Sms is sent the first time from Notes user John to GSM 1234567 the *Auto Matching route* document is created. If the same John sends a second Sms to GSM 1234567 the time stamp is updated.

- Outbound Sms count: when a outbound Sms is sent the first time from Notes user John to GSM 1234567 the *Auto Matching route* document is created. If the same John sends a second Sms to GSM 1234567 the counter is incremented by one.

Specific settings for *Fixed route* documents

Fixed routing may be content based. The content of the SMS message is matched against the specified regular expression. Leave the field empty to suppress content based routing.

Instance Routing Rules
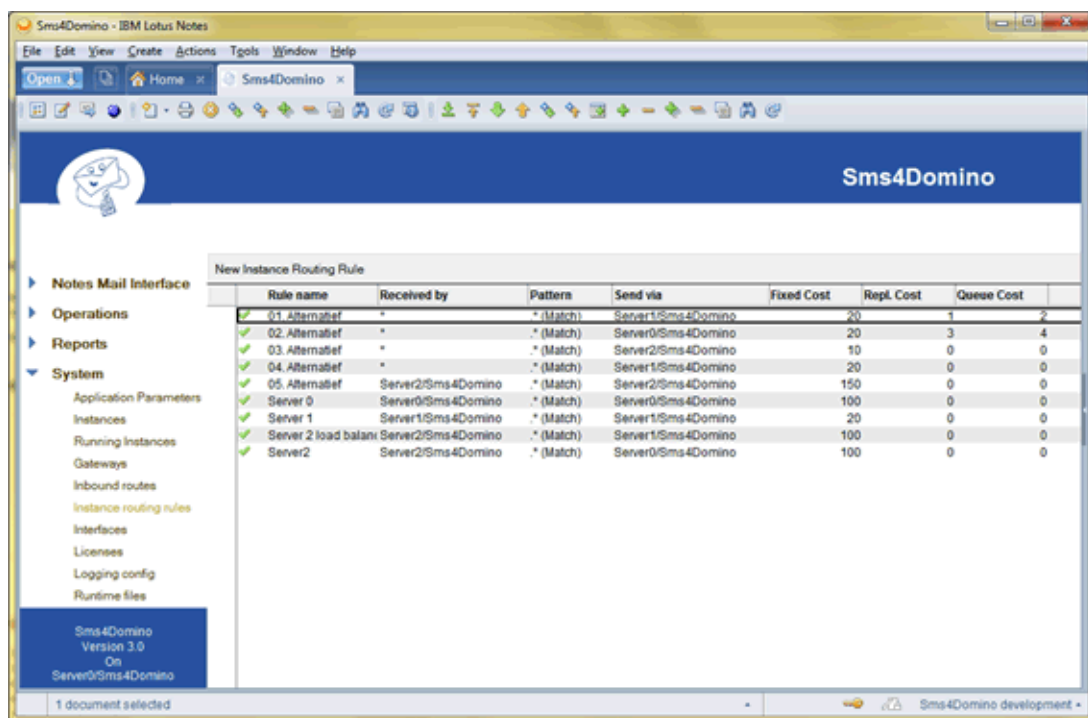
Instance Routing Rule documents are the basis for implementing least cost routing between multiple Sms4Domino instances. But even in a single instance Sms4Domino configuration you must create at least one Instance Routing Rule document.

Examples on configuring least cost routing are described in separate document in this help file.

The settings in the Instance Routing Rules documents are read into memory when Sms4Domino starts. If you need to make changes to any of the settings you will have to quit and start the Sms4Domino application.

Procedure to create a Instance Routing Rule document

- Open the Sms4Domino configuration database.

- Select *System -> Instance routing rules* in the navigator. The right part of the screen displays all defined rules.

  o Only Enabled routes are used by Sms4Domino.

  o Rule name. The rules are sorted alphabetically by name. The name may also appear in the log files. A carefully chosen name will help you during the fine tuning of the least cost routing.

  o Received by - pattern - send via - cost: different parameters for calculating the cost.

Instance Routing Rule document settings

The settings are explained using following example: there are two Sms4Domino instances; one in France and one in Belgium. By preference Sms messages to France should be sent by the France instance; Sms messages to Belgium should be sent by the Belgium instance. Sms messages to other countries may be sent by any available server.

However should one of the servers be overloaded or out of service it is allowed to use the other server.

The configuration on the routing will send the SMS at the lowest cost. The cost is not expressed in read $ or € but in units. The cost is composed of three components:

- The Telco operator cost. Sending an SMS to a local number may cost 10 units but sending an SMS to a foreign country may cost 50 units.

- The replication cost. If a request for sending a SMS is received by the Belgium instance there is a time delay for replicating the request to the France instance. There is some cost related to this delay.

- The queuing cost. If there is large queue of SMS requests at a given instance there will be a delay before the SMS is sent. There is a cost related to the delay.

When a request for sending a SMS is received by the Belgium instance, Sms4Domino will calculate the cost for sending via the local instance and the cost for sending via the France instance.

- Sending via Belgium instance: cost is the sum of the operator cost plus the queuing cost at the local instance. There is no replication cost

- Sending via France instance: cost is the sum of the operator cost plus the queuing cost atthe France instance. On top of that comes the replication cost.

Actual queuing cost is based on the information available in the heartbeat documents. This info is typically outdated by the heartbeat interval plus the replication time.

- Rule name: every rule must have a unique name.

- Status: only active rule documents are being taken into consideration by Sms4Domino

- Condition: when a request for sending a SMS is received by instance x, Sms4Domino will calculate the cost for all Instance Routing Rule documents that have a matching. The condition is composed of following components

  - Sms received by instance: Instance ID of the Sms4Domino instance that receives initially the request for sending the SMS message. Specifying * as instance ID may be used to configure a rule that applies to all Sms4Domino instances.

  - Number pattern: the phone number specified in the outbound SMS request is matched against the specified Java regular expression. The phone number will always be in international format (e.g. +32474964477). A pattern for matching all Sms requests to Belgium would be ^\+32.* (this regular expression means: a string that starts with a plus sign, followed by the digits 3 and 2 followed by an arbitrary number of characters)

  - How to match the number pattern: select from *Phone number should match* or *Phone number should not match* . This provides an easy way to specify a rule for sending SMS messages to any number that is not in Belgium.

- Cost definition: a request for sending a SMS received by theinstance specified in the previous condition may be transmitted via multiple instances. Each instance may have a different cost for sending the Sms. In case there are N Sms4Domino instances there should be N Instance Routing Rule documents for a given condition. The cost is composed of three components

  - Fixed cost: this cost component reflects the fee that must be paid to the Telco operator.

  - Replication cost: this cost is only added to the total cost when de sending instance is different from the receiving instance. It reflects the cost for incurring a delay - due to replication - by sending the SMS.

  - Queuing cost: this cost depends on the active queue on the sending instance. The queue length is stored in the heartbeat records.

- Comment: free text field

Mandatory Instance Routing Rule document for single instance configurations

In a single instance Sms4Domino configuration there is no possibility to do instance routing. Yet, Sms4Domino requires a valid Instance Routing Rule configuration.

- Rule name: LOCALDELIVERY
- Status: Active
- Sms received by instance: *
- Number pattern: .* (dot followed by *)
    - This pattern means "any string composed of 0 or more characters"
- Phone number must match the number pattern
- Cost when transmitted by instance: a wildcard is not allowed. Select the Sms4Domino instance name that is used by the single instance.
- Fixed cost: 1 unit
- Replication cost: 0 units
- Queuing cost: 0 units

Example 1

Instance FRANCE/Sms4Domino

- By preference this instance should send all SMS messages to +33 numbers. The operator cost for sending another country is 10 times higher than sending to France. The cost for sending to France is 5 units.

Instance BELGIUM/Sms4Domino

- By preference this instance should send all SMS messages to +32 numbers. The operator cost for sending another country is 5 times higher than sending to Belgium. The cost for sending to Belgium is 10 units.

It requires four Instance Routing Rule documents.

Document 1	Document 2
Rule name: France_local	Rule name: France_foreign
Sms received by: *	Sms received by: *
Number pattern: ^\+33.* - phone number must match	Number pattern: ^\+33.* - phone number may not match
When transmitted by FRANCE/Sms4Domino	When transmitted by FRANCE/Sms4Domino
Fixed cost: 5 units	Fixed cost: 50 units
Replication cost: 0 units	Replication cost: 0 units
Queuing cost: 0 units	Queuing cost: 0 units
Document 3	Document 4
Rule name: Belgium_local	Rule name: Belgium_foreign
Sms received by: *	Sms received by: *
Number pattern: ^\+32.* - phone number must match	Number pattern: ^\+32.* - phone number may not match
When transmitted by BELGIUM/Sms4Domino	When transmitted by BELGIUM/Sms4Domino
Fixed cost: 10 units	Fixed cost: 50 units
Replication cost: 0 units	Replication cost: 0 units
Queuing cost: 0 units	Queuing cost: 0 units

Suppose Pierre - an employee working in France - deposits his mail for Suzanne@+3345865899@Sms in the Sms4Domino database in France. Sms4Domino France finds 2 rules that match the number pattern: France_local and Belgium_foreign. The first has a cost of 5 units; the other a cost of 100 units. The Sms will be transmitted by the France instance.

Suppose Hilde - an employee working in Belgium - deposits her mail for Jean@+3345855223@Sms in the Sms4Domino database in Belgium. Sms4Domino Belgium finds the same 2 rule documents that match the number pattern: France_local and Belgium_foreign. The first has a cost of 5 units; the other a cost of 100 units. The Sms will be

transmitted by the France instance. This will involve a replication of the request to the France server.

Suppose Jeanne - an employee working in France - deposits his mail for John@+158995255@Sms in the Sms4Domino database in France. Sms4Domino France finds 2 rules that match the number pattern: France_foreign and Belgium_foreign. Both have a cost of 50 units. Sms4Domino knows that replication takes time and server resources. So preference is given to the local instance. The Sms will be transmitted by the France instance.

In this example all Sms messages to a France number will always be transmitted by the France instance, even if the France server is heavily loaded. Example two will focus on load balancing.

Example 2

In this example we focus on load balancing. There are two instances, both located in Belgium. The instances are part of a Domino cluster; replication is within seconds.

As both instances are located in Belgium they have the same operator cost for sending messages.

We would like to try to balance the load over the two servers.

Instance INSTANCEA/Sms4Domino

- This instance should send all SMS messages that are posted by the router in the local configuration database. However, if the local queue contains more than 60 requests than the remote queue the request should be handled by the other instance. (A queue of 60 requests results in a delay of about 10 minutes)

Instance INSTANCEB/Sms4Domino

- Same behaviour as INSTANCEA

It requires two Instance Routing Rule document.

Document 1	Document 2
Rule name: A	Rule name: B
Sms received by: *	Sms received by: *
Number pattern: .* - phone number must match	Number pattern: .* - phone number must match
When transmitted by INSTANCEA/Sms4Domino	When transmitted by INSTANCEB/Sms4Domino
Fixed cost: 5 units	Fixed cost: 5 units
Replication cost: 60 units	Replication cost: 60 units
Queuing cost: 1 unit/Sms	Queuing cost: 1 unit/Sms

End users send mail to name@number@sms. We assume the Domino router deposits the mails in the configuration database of InstanceA.

Suppose the first request for sending a SMS is received. InstanceA finds 2 rules that math the number pattern. Rule A has a cost of 5 units. Rule B has a cost of 5 + 60 = 65 units. The Sms will be transmitted by InstanceA.

Suppose that more requests arrive. At a certain moment the queue of InstanceA contains 40 requests. When a new request arrive rule A evaluates to a cost of 5 + 40*1 =45 units. Rule B has a total cost of 65 units. So InstanceA will still be used.

When the queue length becomes higher than 60 the resulting cost of rule A becomes 5 + 61*1 = 66 units. Rule B has a cost of 65 units. This is lower. The request replicates to InstanceB.

Should the router deposit a request in the config database of InstanceB, it is InstanceB that calculates the best route. Rule A has a cost of 5 + 60 + 61*1 = 126 units. Rule B has a cost of 5 units. Instance B will be used.

Remark: the queuing cost is based on the queue length that is published in the heartbeat records. Sms4Domino tries to keep the info up to date but the info will always be slightly outdated.

Configuration of Domino mail topology

End users will mostly use the Notes mail interface to send Sms messages. They compose a mail message using the Notes client or iNotes and address it to

John@0474778899@Sms

The @ character is recognized by the Domino router as a Domino domain separator. Domino must know how to route the mail to the foreign domain called Sms. The Sms4Domino configuration database must be specified as the gateway server and file.

In case your Domino environment consists of multiple Domino servers you may need to configure Domino connection documents to specify the path from each server to the server that holds the Sms4Domino configuration database.

We will first show how to configure Domino mail routing for a Single Instance topology and then for the more complex Multiple Instance topology.

Following text has been copied from the Lotus Administrator Help File

IBM® Lotus® Domino® treats external messaging applications, such as fax or pager gateways, as foreign domains. To route mail from a Domino domain to an external application, create a Foreign domain document.

Creating a Foreign domain document

A Foreign domain document defines the path between a Domino domain and an external application, such as a fax or pager gateway. A Foreign domain document identifies the Domino server that acts as the gateway to the external application.

Although Foreign domains are mostly used for third party applications, you can also use them to transfer messages between a Release 5.0 or later server and a Release 3.x SMTP server.

Restrictions that you set on this Foreign domain document apply only to the From domain of the previous hop. These restrictions work in conjunction with those in the Configuration Settings document. Domino always defaults to the most restrictive entry.

To create a Foreign domain document

1. From the Domino Administrator, click the Configuration tab and then expand the Messaging section.
2. Choose Domains.
3. Click Add Domain to create a new Domain document.
4. Click the Basics tab, and complete these fields:

Field	Enter
Domain type	Choose Foreign domain.
Foreign Domain Name	The domain name of the foreign mail system. This name was chosen when the MTA or gateway was installed.
Domain description	An optional description of the gateway or MTA.

5. Click the Restrictions tab, and then complete these fields:

Field	Enter
Allow mail only from domains	The names of Domino domains that are allowed to route messages to this foreign domain. Leave this field blank to allow any domain to route mail through the local domain to the foreign domain.

| Deny mail from domains | The names of Domino domains that are not allowed to route messages to this foreign domain. Leave this field blank to allow any domain to route mail through the local domain to the foreign domain. |

6.   Click the Mail Information tab and complete these fields, and then save the document:

Field	**Enter**
Gateway server name	The name of the Domino server running the gateway software.
Gateway mail filename	The gateway's mail file name. See the documentation that came with the gateway for the proper file name.

7.   Create a Connection document to specify how servers in the current domain connect to the foreign domain.

Single Instance Sms 4Domino topology

In this example the Sms4Domino configuration is located on server Server 1/Sms4Domino.

End users will send a mail to *name@number@SMS* in order to send a SMS message

- Create a foreign domain document in the Domino directory
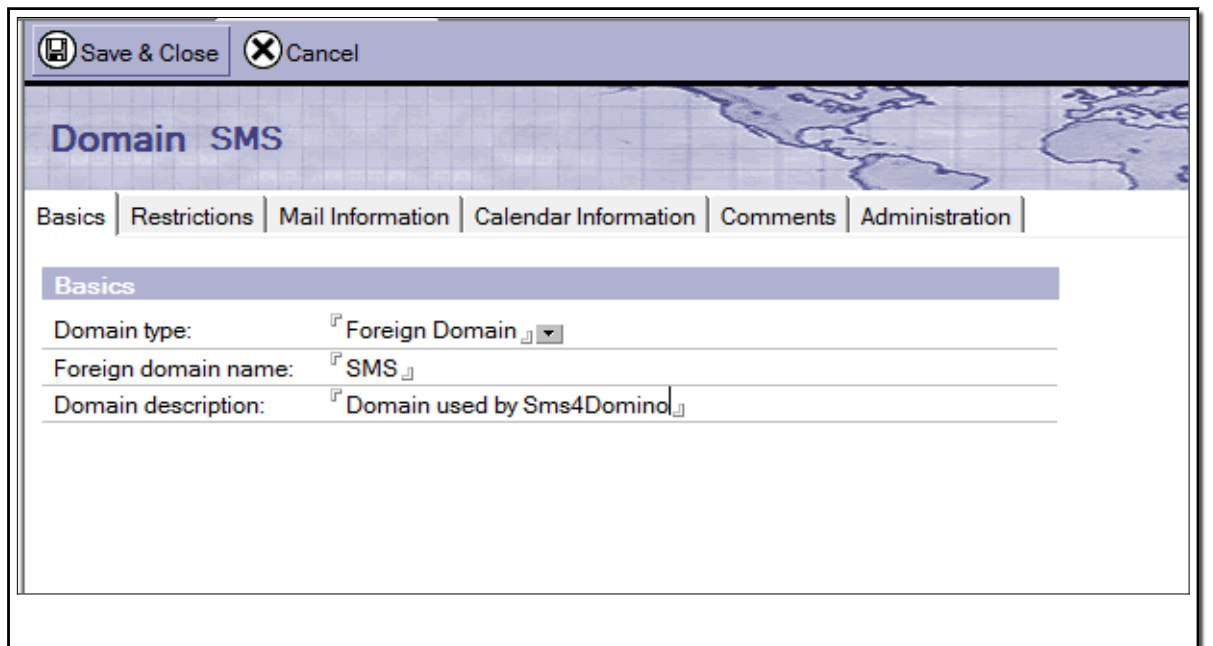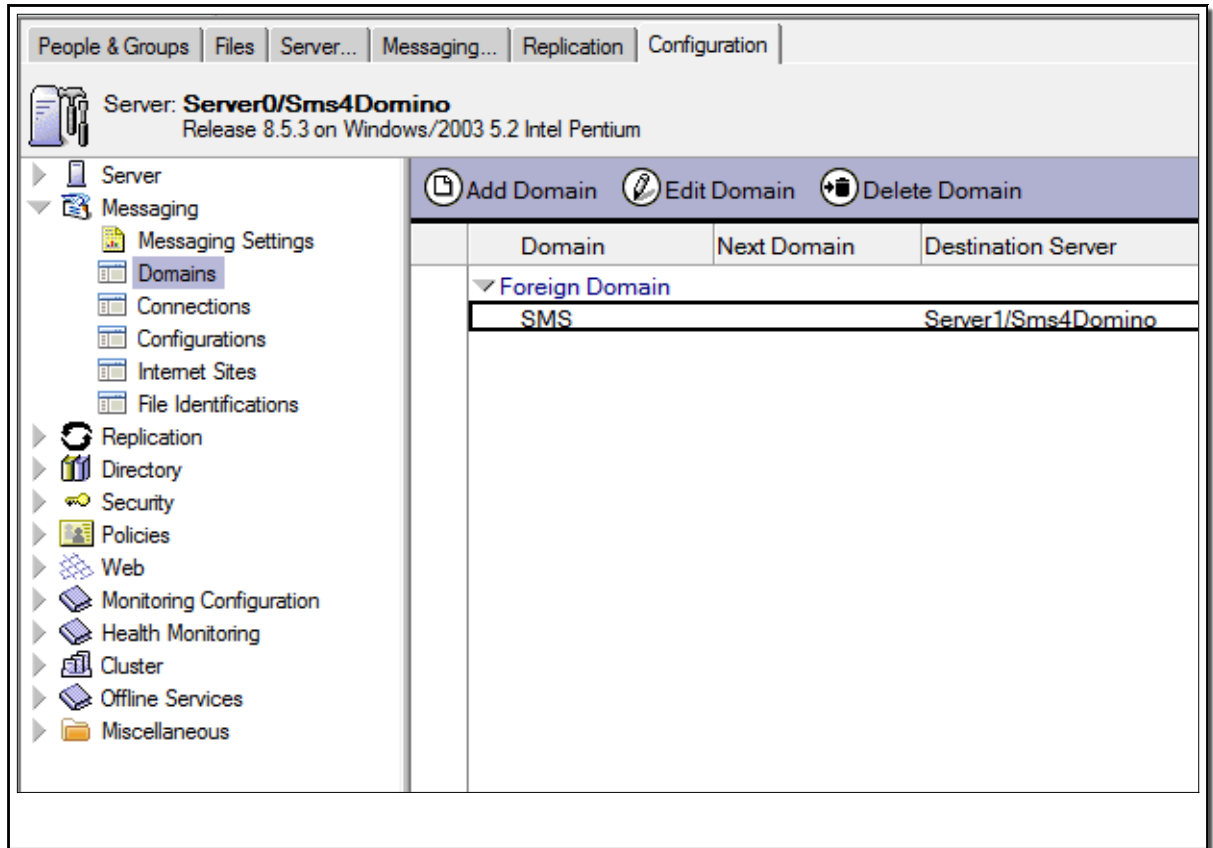
- If needed, create connection documents to route mail from Server0/Sms4Domino to Server1/Sms4Domino and another connection document for routing in the opposite direction.

Multiple Instance Sms 4Domino topology

In this case there is more than one Sms4Domino configuration database. This may be for improved reliability or to benefit from least cost routing.

Case 1: two clustered Domino servers in the same country

The communication costs are the same for both servers. End users should not know that there is more than one Sms4Domino server. They just send their mail to *name@number@SMS* in order to send a SMS message.

There should be two foreign domain documents for the domain SMS. One points to the first server, the other to the second server. The Domino router will deposit the mail in one of the two databases.

● Create two foreign domain documents for the same foreign domain name in the Domino directory

- ~~Clustered Domino servers reside in the same Notes Named Network. No connection~~ documents are required for mail routing.

Case 2: two Domino servers in the different countries

We could implement the same solution as in case 1. However SMS messages for France could be deposited by the router in the Sms4Domino config database in Belgium.

As an alternative we could train the users located in France to send their mail to *name@number@SMSFrance* in order to send a SMS message. The users located in Belgium are instructed to send their mail to *name@number@SMSBelgium* in order to send a SMS message.

We need two foreign domain documents: one for the domain SmsFrance that points to the Sms4Domino configuration database located in France and one for the domain SmsBelgium that points to the Sms4Domino configuration database located in Belgium.

It is also possible to combine the two above solutions by creating 4 foreign domain documents: two for SMS, one for SmsFrance and one for SmsBelgium.

Logging settings

Sms4Domino makes use of the open source library log4j for all logging. This framework provides a very flexible and performant logging system. Plenty of background information is available on the Internet. We recommend you to visit the web site http://logging.apache.org/log4j/1.2/.

As an administrator you have almost full control over what info should be logged and what should be the output for the logging. We will describe some typical configurations in this help document.

The configuration of log4j is done via properties files (text files comparable to notes.ini). Sms4Domino makes use of Notes documents to define the configuration of the logging. These documents are read by Sms4Domino and converted into log4j properties.

Each Sms4Domino instance requires at least one logging configuration document.

The log settings may be modified at any time. The Sms4Domino command *refresh logging* reads the updated configuration from the configuration database and modifies the active log4j settings accordingly.

Some log4j concepts

- logger: applications that use log4j direct all logging output to one or more named loggers. The logger name space forms a hierarchical system that starts at the top with the root logger. When an application produces some log info it specifies the name of the logger and the severity (or log level) of what is being logged. Possible log levels are
  - FATAL
  - ERROR
  - WARN
  - INFO
  - DEBUG
  - TRACE
- appender: log4j directs the output of the log statements to one or more appenders that have been defined for the logger. Typical appenders write the logging info to the console, a file, the Windows event manager, Domino logging, ... More than one appender may be linked to a logger. Multiple loggers may share the same appender.
- layouts: the events logged by an application may contain plenty of information: time and date of the event, Java class name, method name and line number where the event occurred, text of the event, ... Layouts give you the possibility to format the output. In all following examples we will use the standard PatternLayout.

Example log4j configuration file

Let us first have a look to a standard configuration file.

```
log4j.rootLogger=INFO, console1
log4j.appender.console1=org.apache.log4j.ConsoleAppender
log4j.appender.console1.encoding=UTF-8
log4j.appender.console1.layout=org.apache.log4j.PatternLayout
log4j.appender.console1.layout.ConversionPattern=%d{HH:mm:ss,SSS} %m   %n
```

The first line configures the root logger. All log events of level INFO should be written to the appender called console1. Automatically all log events of level WARN, ERROR and FATAL will also be logged to the appender. Log events of level DEBUG and TRACE will not be routed to the appender.

All loggers form a hierarchy. The root logger is sitting at the top. As we enabled the root logger all other loggers will automatically be enabled for the INFO, WARN, ERROR and FATAL level.

The second line configures the appender called console1. The appender is based on a ConsoleAppender. This is a standard appender that writes all output to the console. In a

Sms4Domino environment the console is either the Domino console and log.nsf (when Sms4Domino is running as an add-in) or the Sms4Domino Swing console (when Sms4Domino is running as a standalone program).

The third line is required to propertly format no Ascii characters.

The fourth line specifies how the output should be formatted. We are using a standard PatternLayout. This layout requires additional information that is supplied on the fourth line.

The fifth line specifies what information should be logged for each event. In the example

- %d{HH:mm:ss,SSS}: time of the event in the format HH:mm:ss,SSS
- %m: the message supplied by the application
- %n: new line character

Procedure to create a Logging Configuration document

- Open the Sms4Domino configuration database.

- Select *System -> Logging config* in the navigator. The right part of the screen displays all defined logging configuration document.

   o Only Active logging configuration documents are used by Sms4Domino.

   o Sms4Domino will look for logging configuration documents that correspond to the user-id that is running Sms4Domino. Sms4Domino will build a log4j properties file that is the concatenation of all active logging configuration documents for the current Notes user.

   o The title is just for documentation
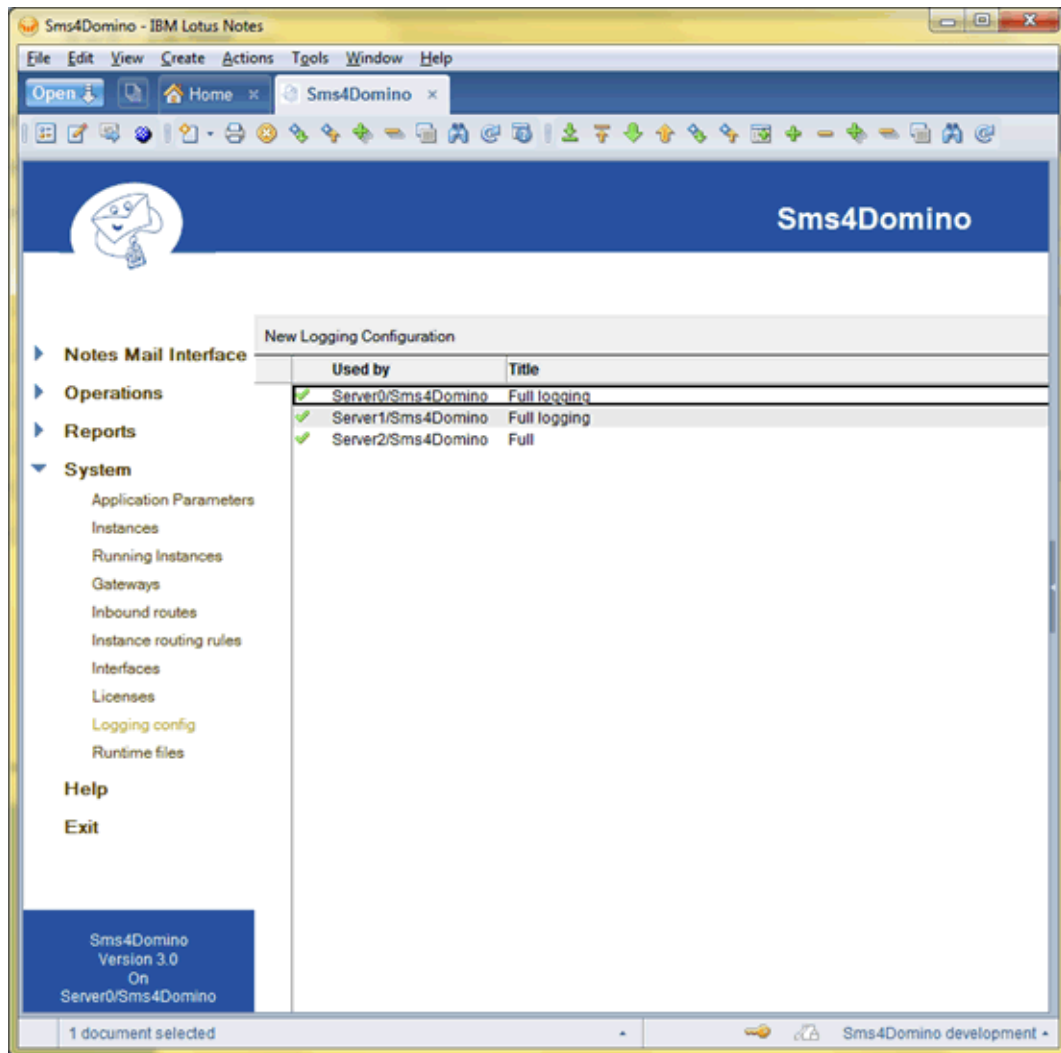
- Click on the action button *New Logging Configuration* or double click on an existing Logging Configuration document.

Logging Configuration settings

- Used by Instance: every running Instance requires a specific logging configuration document
- Title: just for documentation
- Status: All active logging configuration documents for the user are concatenated into a single properties file. The resulting properties file is then used as the log4j.properties file. It is possible to change the logging parameters whilst the SMS4Domino server is running. Use the command *refresh logging* to reload the logging config.
- Log4j.properties: log4j settings as they would appear in a log4j properties file. Text starting with a # character is considered being a comment.
  The action button *Set default for standard logging* configures log4j for using a single console appender that outputs all INFO level events to the Sms4Domino console.
- Click *Save and Close* to save your changes.

Sms4Domino Help

Conversion Pattern

The conversion pattern specifies what information must be logged for an event. Example

> log4j.appender.console1.layout.ConversionPattern=%d{HH:mm:ss,SSS} %m   %n

The conversion pattern is written to the right of the = sign. It is composed of literal text and conversion specifiers. Conversion specifiers start with the % character followed by optional format modifiers and a conversion character. The conversion character specifies the type of data, e.g. category, priority, date, thread name. The format modifiers control such things as field width, padding, left and right justification. The following is a simple example copied from the log4j Javadoc (http://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/PatternLayout.html)

Let the conversion pattern be "%-5p [%t]: %m%n" and assume that the log4j environment was set to use a PatternLayout. When logging the messages "Message 1" and "Message 2" would yield the output

> DEBUG [main]: Message 1
> WARN [main]: Message 2

Some conversion characters

- %c: name of the logger
- %d: date of the logging event. It may be followed by a date format written between { }. E. g. %d{HH:mm:ss,SSS} or %d{dd MMM yyyy HH:mm:ss,SSS}. If no date format specifier is given then ISO8601 format is assumed.
- %m: the application supplied message associated with the logging event.
- %n: the platform dependent line separator character or characters
- %t: name of the thread
- %%: the sequence %% outputs a single percent sign.

Console Appender settings

A console appender should be used to route all logging information to the Sms4Domino console. When Sms4Domino runs as a server add-in the console is integrated with the Domino console and the Domino log.nsf.

A typical console appender is configured with following four configuration entries (assuming the appender is called console1 as defined by *log4j.rootLogger=INFO, console1*)

```
log4j.appender.console1=org.apache.log4j.ConsoleAppender
log4j.appender.console1.layout=org.apache.log4j.PatternLayout
log4j.appender.console1.encoding=UTF-8
log4j.appender.console1.layout.ConversionPattern=%d{HH:mm:ss,SSS} %m   %n
```

File Appender settings

File appenders are used to route all logging information to one or more files.

- FileAppender appends log events into a file. The file is specified by the File option. If the file already exists, it is either appended or truncated depending on the value of the Append parameter.
- RollingFileAppender acts as a FileAppender but will create a new file when the log file reaches a certain size specified by the MaxFileSize parameter. When creating a new file the current log file is renamed by appending a .1 to the filename. Should this .1 file already exist it is renamed by changing the .1 to .2. The maximum number of files to keep is specified by the MaxBackupIndex parameter.
- DailyRollingFileAppender acts as a FileAppender but will create a new file at a user chosen interval. The file names will contain the date information.

A typical configuration file for using a FileAppender (assuming the appender is called file1 as defined by *log4j.rootLogger=INFO, console1, file1*)

```
log4j.appender.file1=org.apache.log4j.FileAppender
#use \\ or / as directory separator when specifying the file name
log4j.appender.file1.File=e:\\folder\\example.log
# Append parameter: if true the logger will append to the file should it already exist
log4j.appender.file1.Append=true
```

A typical configuration file for using a RollingFileAppender (assuming the appender is called file1 as defined by *log4j.rootLogger=INFO, console1, file1*)

```
log4j.appender.file1=org.apache.log4j.RollingFileAppender
#use \\ or / as directory separator when specifying the file name
log4j.appender.file1.File=e:\\folder\\example.log
# Append parameter: if true the logger will append to the file should it already exist
log4j.appender.file1.Append=true
# The MaxBackupIndex option determines the number of previously rolled files to preserve.
# If set to zero, then no roll over occurs and the log file is simply truncated when it reaches MaxFileSize.
log4j.appender.file1.MaxBackupIndex=5
# The MaxFileSize option takes a String value representing a long integer in the range 0 - 263. You can specify the value with the suffixes "KB", "MB" or "GB" so that the integer is interpreted as being
# expressed respectively in kilobytes, megabytes or gigabytes.
# Rollover occurs when the log file reaches MaxFile-Size. One log event will always be contained within one file
log4j.appender.file1.MaxFileSize=100MB
```

A typical configuration file for using a DailyRollingFileAppender (assuming the appender is called file1 as defined by *log4j.rootLogger=INFO, console1, file1*)

```
log4j.appender.file1=org.apache.log4j.RollingFileAppender
#use \\ or / as directory separator when specifying the file name
log4j.appender.file1.File=e:\\folder\\example.log
# Append parameter: if true the logger will append to the file should it already exist
log4j.appender.file1.Append=true
# The DatePattern specifies when the rollover should occur. The parameter should follow the conventions of the java.text.SimpleDateFormat class
#   yyyy for yearly rollover
#   yyyy-MM for monthly rollover
#   yyyy-MM-dd for daily rollover
#   yyyy-MM-dd-a for daily rollover at midnight and midday
#   yyyy-MM-dd-HH for hourly rollover
# The pattern is used to specify when the rollover should occur and what suffix should be
```

added to the filename. The pattern may include single quoted text. This may be usefull for specifying the file suffix
log4j.appender.file1.DatePattern='.'yyyy-MM-dd-HH

Using multiple appenders

When running Sms4Domino as a server addin it may be sufficient to use one ConsoleAppender.
All historical logging will be stored in log .nsf automatically.
However, when Sms4Domino runs as a standalone program, you may want to use a
ConsoleAppender and a FileAppender. This is accomplished by specifying more than one
appender for the rootLogger

    log4j.rootLogger=INFO, console1, file1

The level of logging (INFO in the above example) is always the same for all specified appenders.

Should it be required to have a different level of logging for the console and file appenders you
could specify the most verbose level of logging for the rootLogger and then filter the logging at the
appender level.
In following example we would like DEBUG level logging to a file and INFO level logging to the
console. This is achieved by specifying the Threshold parameter for the least verbose appender

    log4j.rootLogger=DEBUG, console1, file1
    log4j.appender.console1.Threshold=INFO

Server license settings

All license settings are found via *System -> Licenses*.

Sms4Domino looks at regular intervals for the presence of a valid license. Each running istance requires a specific License document. License information is stored, as a signed human readable text string, in a license document. The Sms4Domino configuration database could hold multiple license documents (in order to support multi instance deployments).

The validity of a license may depend on multiple factors. Typical elements are

- Servername where the configuration database is stored

- Instance ID of the running instance

- User-id that runs Sms4Domino

- Number of active inbound or outbound gateways defined by the configuration database

- Expiration date of the license. The current date must be before this expiration date

- Release date of Sms4Domino. The release date of the running Sms4Domino program must be before this release date

- Number of Notes Mail Interfaces defined by the configuration database

- Number of Test gateways defined by the configuration database.

A typical evaluation license that could be downloaded from the website grants everybody the right to run Sms4Domino with one Test Gateway and one Notes Mail Interface. Every Sms message will contain the text "Sms4Domino" in addition to the user generated text.

Procedure to create a license document

- Open the Sms4Domino configuration database.
- Select *System -> Licenses* in the navigator. The right part of the screen displays all defined license documents
  - o Only Active license documents are used by Sms4Domino. They have a green check mark
  - o Servername: when the configuration database is located on server x then Sms4Domino looks for a license docuemnt for servername x
  - o Used by: the Instance ID that will use the license settings.
  - o License data: indication if a license file has been loaded into the document

.
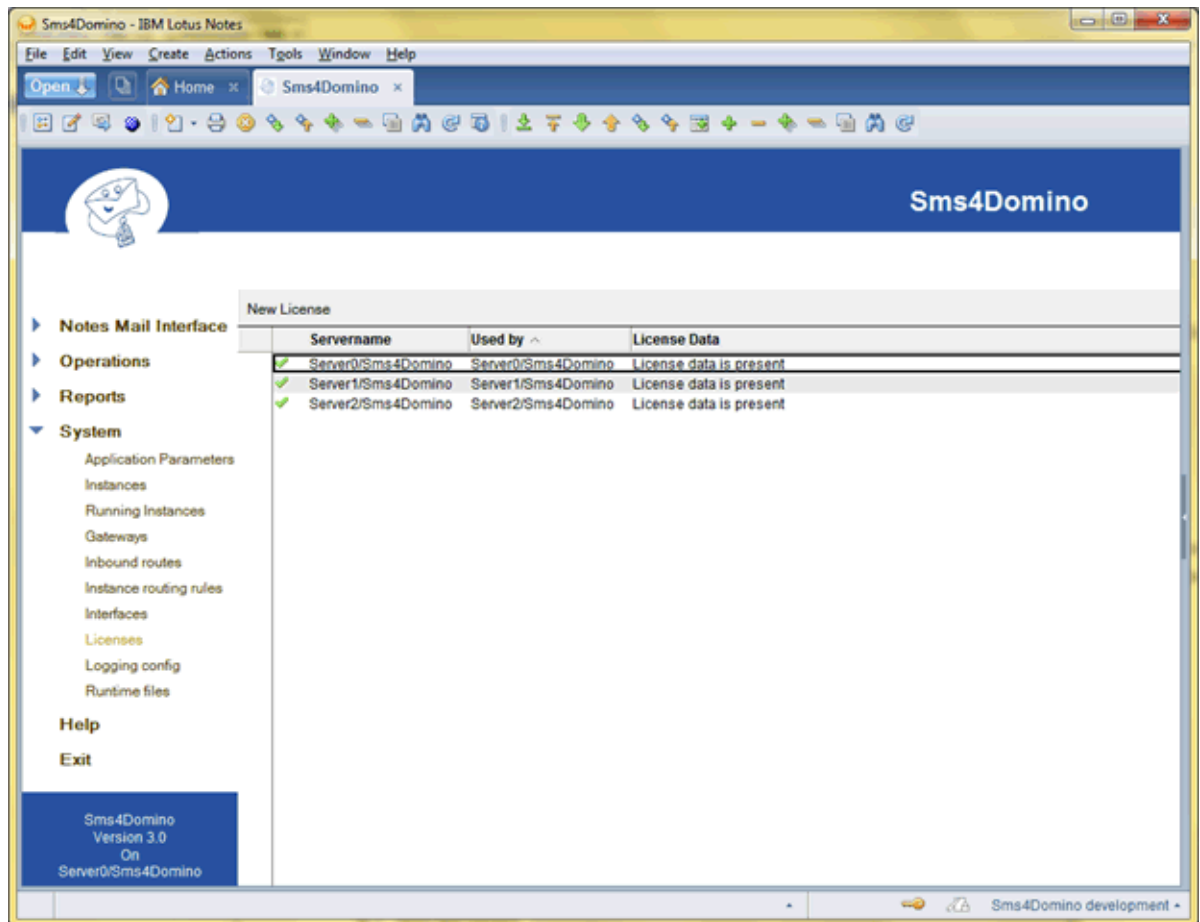
License document settings

- Servername - Instance Id: Sms4Domino looks for an active license document that corresponds to the servername where the configuration database is stored and to the Instance Id that runs the Sms4Domino code. The Instance Id is specified in Notes.ini. When missing it is either the Domino servername (when running as a server addin program) or the username linked to the userid of the Notes client installed on the workstation (when running as a standalone program on a workstation).

- Status: active or not used. There should only be one active license document for a given servername/username combination

- License data: contents of the license file that was received at time of purchase. The license file may be opened with a text editor. However, changing a single character will invalidate the signature. License files without a valid signature will not be accepted by Sms4Domino. The license file may contain wildcard settings (based on Java regular expressions). When you receive the license file you must open it with a text editor (e.g. Notepad), select all text and paste the text in the field.
  Text before the ***START LICENSE*** header or after the ***END LICENSE*** footer is ignored during the validation of the license.

License - IBM Lotus Notes

File Edit View Create Actions Tools Window Help

Open | Home × | Sms4Domino × | License ×

Close Edit Document

License

SMS4Domino server requires a valid license file present in the configuration database. The signed license file authorizes the user to run the program for a given configuration. The license file is obtained via the web site http://www.sms4domino.eu. Demo licenses are available.
Multiple license documents may be present in the database. At startup, SMS4Domino will look for a matching license document.

Servername	Server0/Sms4Domino	Specify the servername where the configuration database is located.
Instance Id	Server0/Sms4Domino	Specify the Instance Id of the effective Notes user that runs the Sms4Domino program. The Instance Id is specified in Notes.ini. When missing it is either - When running as a server addin program this is the servername. - When running as a standalone program on a workstation this is the username linked to the userid of the Notes client installed on the workstation.
Status	Active	Only active licenses are taken into account when SMS4Domino starts.
License data	***START LICENSE*** Servername=.* InstanceId=Server.?/Sms4Domino Expires=31/12/2012 ReleaseDateLimit=31/12/2012 MaxInGateways=3 MaxOutGateways=4 AllowTestInGateway=true AllowTestOutGateway=true Evaluation=true MaxNotesMailInterfaces=3 ***START SIGNATURE*** 302c02144e2a60e7f7653f007e593df1e932ab8c42e4 c25c0214585be19953eb1aa6034f383be7e030e6bbc b3c08 ***END LICENSE***	The complete contents of the license file must be pasted in this field. Modifying the contents of the file will invalidate the electronic signature.
Comment		For your internal use.

Sms4Domino development

How to amend a license ?

License files may contain an expiration date. At some point in time it could be necessary to renew the license.

Procedure

1. Sms4Domino runs based on the settings of the current active license document

2. A new license document may be configured for the same user and server. The status of this document should be "Inactive"

3. When the new license file has been received and loaded into the license document the administrator should

    *   Modify the status of the current active license document to "Inactive". Save and close the document.

    *   Modify the status of the newly created license document to "Active". Save and close the document.

    *   Restart Sms4Domino

Error messages during license validation

When Sms4Domino is started the license is validated. In case of a missing license following messages are logged on the Domino console

> 21:12:16,580 SmsServer is ready to be configured and started. TRACE
> 21:12:16,586 Current configuration violates license conditions : 0 active license documents found for user-server XXXX-YYYY. Should be 1.   FATAL
> 21:12:16,588 SMS4Domino is NOT started   FATAL
> Application will close in 15 seconds.

In case of an error in the signature the logging becomes

> 21:15:30,894 SmsServer is ready to be configured and started. TRACE
> 21:15:31,016 Unexpected error during validation of signature. Stack trace follows: ERROR
> java.security.SignatureException: invalid encoding for signature :java.io.IOException: Sequence tag error
>   at com.ibm.crypto.provider.SHA1withDSA.engineVerify(Unknown Source)
>   at java.security.Signature$SignatureImpl.engineVerify(Signature.java:467)
>   at java.security.Signature.verify(Signature.java:287)
>   at
> be.sms4domino.licensing.LicenseSignature.isValidSignature(LicenseSignature.java:88)
>   at
> be.sms4domino.licensing.License.loadActiveLicenseFromDomino(License.java:174)
>   at be.sms4domino.server.DominoSmsServer.<init>(DominoSmsServer.java:64)
>   at be.sms4domino.server.DominoSmsServer.getInstance(DominoSmsServer.java:76)
>   at Sms4DominoStandalone.runNotes(Sms4DominoStandalone.java:154)
>   at lotus.domino.NotesThread.run(Unknown Source)
> 21:15:31,017 Current configuration violates license conditions : Signature in license data is invalid. User-server XXXX-YYYY   FATAL
> 21:15:31,018 SMS4Domino is NOT started   FATAL
> Application will close in 15 seconds.

The license will be checked at regular times to make sure the expiration date has not been passed.

Principles of operations

Sms4Domino may run as a Domino server addin or as a standalone program.

End users compose Sms messages by sending an e-mail to e.g; John@0474876212@sms. This can be done from the Notes clients, iNotes®, Lotus Traveler®, ... The Domino router knows how to deliver the message to the Sms4Domino configuration database. Sms4Domino picks up the mail, converts it into a SMS message and transmits it via one of the gateways.

Of course it is also possible to create the mail message from non-Notes environments. For example a server monitoring program may detect a server failure and send a SMTP mail message to one of the Domino servers. In this case the mail must be addressed to John%0474876212%sms@company.com. The SMTP listener task accepts the message. The part before the @company.com is interpreted as a Lotus Notes® mail address where % is used as a Domino domain separator.

End users may ask for a delivery report. When Sms4Domino receives the delivery report from the GSM network, it is converted into a regular mail and routed back to the end user.

In the following documents you will learn how to start Sms4Domino and how to monitor the operations.

Run Sms4Domino as a Domino server addin

Java programs may be executed under control of the RunJava task. The Java Virtual Machine that is shipped with Lotus Domino® (typically stored in a subdirectory of IBM\Lotus\Domino\jvm) must be used, as Sms4Domino depends on some cryptographic API's that are shipped by IBM® and are not part of Oracle's® jvm.

The run-time files must have been correctly detached to the local file system. You should strictly follow the installation instructions. Otherwise you will see security exceptions in the log file.

Sms4Domino is started by entering the Domino server command

    load runjava Sms4DominoAddin

As a shortcut the following command produces the same result

    load runjava Sms

The name Sms4DominoAddin or Sms is case sensitive.

Sms4Domino commands are entered by typing

    tell Sms <command>

Sms4Domino may be stopped by entering the quit command. However, if other tasks are running

under control of RunJava, not all Java objects will be removed from memory . It is better to stop Sms4Domino by issuing the *tell runjava quit* command.

Sms4Domino may be launched when Domino starts by modifying the notes.ini file of the server. Look for the entry ServerTasks and add *RunJava Sms* at the end

      ServerTasks=LDAP, router, RunJava Sms

Run Sms4Domino as a standalone program

Sms4Domino may run as a standalone program. This Java program makes local Domino calls. The computer must satisfy following conditions

- Lotus Notes® 8.x client must have been installed on the computer.
- Notes.ini of the Notes client must contain a valid entry *keyfilename=<filepath of the user.id file>* . It is recommended to test the Notes setup by launching the Notes client and opening the Sms4Domino configuration file on the server.
- The run-time files must have been correctly detached to the local file system. You should strictly follow the installation instructions. Otherwise you will see security exceptions in the log file.

Sms4Domino may be launched by executing following batch file. The name Sms4DominoStandalone is case sensitive. We assume that Lotus Notes® has been installed in c:\program files (x86)\IBM\Lotus\Notes

```
c:
cd \program files (x86)\IBM\Lotus\Notes\jvm\bin
java Sms4DominoStandalone
```

In case the user.id file is protected by a password you will be prompted to enter the password. It is also possible to specify the password on the command line as a parameter

```
c:
cd \program files (x86)\IBM\Lotus\Notes\jvm\bin
java Sms4DominoStandalone password
```

The Sms4Domino console will be launched.
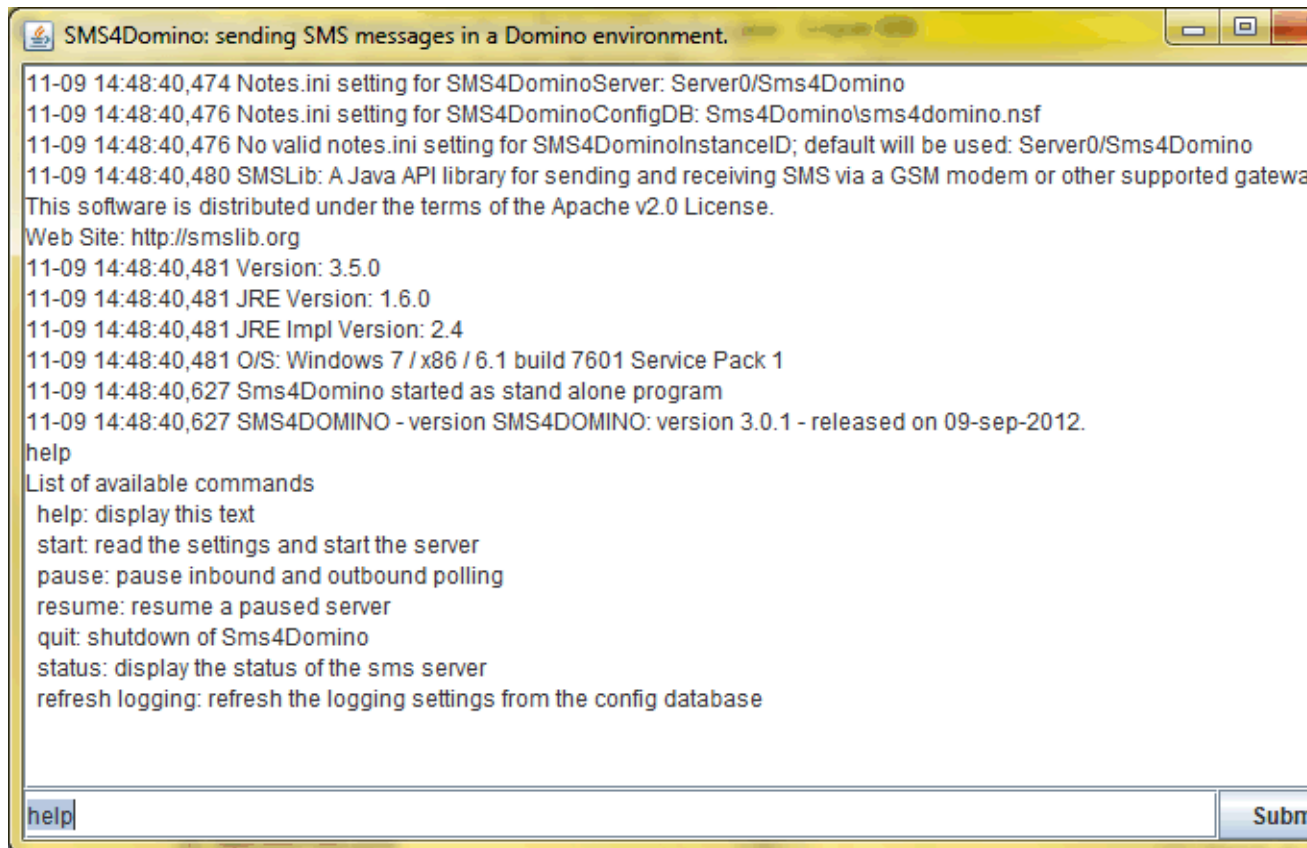
The bottom area is used to enter commands. Use the arrow-up and arrow-down keys to scroll through the command history.

The Windows may be resized. Sms4Domino stores the window size in the notes.ini.

Sms4Domino is terminated by entering the quit command. Should this fail - and this could happen when some low level I/O cannot be interrupted - the program is stopped by clicking the x in the top right corner of the console window.

Sms4Domino commands

When Sms4Domino has been launched the program is ready to accept commands entered via the console. Sms4Domino is not yet ready to send SMS messages unless the autostart parameter was specified in the application profile.

The Sms4Domino console keeps a history of entered commands. Use the up and down arrow to scroll through the list.

<u>Help command</u>

Syntax: help
Purpose: displays the list of known commands

SMS4Domino: sending SMS messages in a Domino environment.

```
help
List of available commands
  help: display this text
  start: read the settings and start the server
  pause: pause inbound and outbound polling
  resume: resume a paused server
  quit: shutdown of Sms4Domino
  status: display the status of the sms server
  refresh logging: refresh the logging settings from the config database
```

<u>Status command</u>

Syntax: status
Purpose: displays the status of Sms4Domino. This status is also displayed in the gateway document (see the help document about gateway documents)

SMS4Domino: sending SMS messages in a Domino environment.

```
  help: display this text
  start: read the settings and start the server
  pause: pause inbound and outbound polling
  resume: resume a paused server
  quit: shutdown of Sms4Domino
  status: display the status of the sms server
  refresh logging: refresh the logging settings from the config database
status
Server status: STOPPED
```

<u>Start command</u>

Syntax: start
Purpose: starts a stopped Sms4Domino server. Once started, the server will be processing requests for sending Sms messages

Remark: it is recommended to configure Sms4Domino in auto-start mode. This avoids having to enter the start command.

```
SMS4Domino: sending SMS messages in a Domino environment.
Server status: STOPPED
start
javax.comm.SerialPort implementation: version 3.8
Copyright (c) 1998-2009 Serialio.com, All Rights Reserved.
Serialio Library: version 10.1.0: build 9218
Copyright (c) 1996-2011 Serialio.com, All Rights Reserved.
os.name="Windows 7"  os.arch="x86"
SerialPort class loaded: jspWin
17:39:18,053 Effective smslib settings:
17:39:18,053  SERIAL_NOFLUSH:false
17:39:18,053  SERIAL_NOEVENTS:false
17:39:18,053  SERIAL_POLLING:false
17:39:18,054  SERIAL_POLLING_INTERVAL:200
17:39:18,054  SERIAL_TIMEOUT:30000
17:39:18,054  SERIAL_KEEPALIVE_INTERVAL:120
17:39:18,054  SERIAL_BUFFER_SIZE:16384
17:39:18,054  SERIAL_CLEAR_WAIT:1000
17:39:18,054  SERIAL_RTSCTS_OUT:true
17:39:18,054  QUEUE_RETRIES:0
17:39:18,054  AT_WAIT:200
17:39:18,054  AT_WAIT_AFTER_RESET:10000
17:39:18,054  AT_WAIT_CMD:1100
17:39:18,054  AT_WAIT_CGMS:200
17:39:18,054  AT_WAIT_NETWORK:15000
17:39:18,054  AT_WAIT_SIMPIN:10000
17:39:18,055  AT_WAIT_CNMI:3000
17:39:18,055  OUTBOUND_RETRIES:3
17:39:18,055  OUTBOUND_RETRY_WAIT:3000
17:39:18,055  WATCHDOG_INTERVAL:15
17:39:18,055  CNMI_EMULATOR_INTERVAL:30
17:39:18,055  MASK_IMSI:false
17:39:18,055  DISABLE_CMTI:true
17:39:18,055  HOURS_TO_ORPHAN:72
17:39:18,055  CONCURRENT_GATEWAY_START:true
17:39:18,055  DISABLE_CMMS:true
17:39:18,055  DISABLE_COPS:false
17:39:18,055  CACHE_DIRECTORY:java.home
17:39:18,055  QUEUE_DIRECTORY:
17:39:18,057 No outbound gateway has been configured.
17:39:18,090 GTW: NOKIA6070: Starting gateway, using Generic AT Handler.
17:39:18,090 GTW: NOKIA6070: Opening: COM6 @115200
javax will be used
17:39:26,636 GTW: NOKIA6070: GSM: Registered to home network.
17:39:27,436 GTW: NOKIA6070: MEM: Storage Locations Found: MESM
17:39:27,437 GTW: NOKIA6070: Callback indications will be disabled and CNMI emulator will be enabled.
17:39:28,636 GTW: NOKIA6070: Signal level/bit error rate: +CSQ: 31,99OK
17:39:29,436 GTW: NOKIA6070: Network registration: +CREG: 1,1OK
17:39:29,836 GTW: NOKIA6070: Network operator: +COPS: 0,2,"20601"OK
17:39:29,838 GTW: NOKIA6070: Gateway started.
SMS server successfully started
```

Pause command

Syntax: pause
Purpose: a started server may be paused. It will halt the polling of the interfaces.
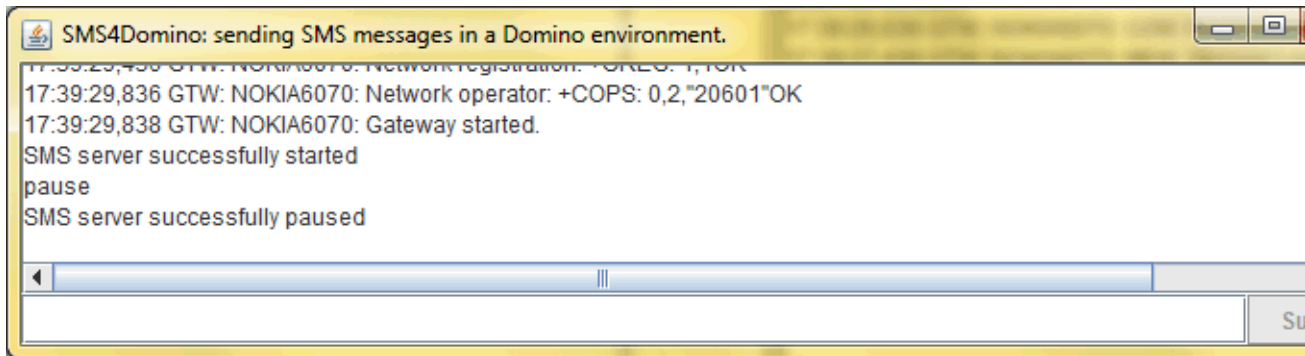
Resume command

Syntax: resume
Purpose: a paused server may be resumed. It will continue to process all Sms requests
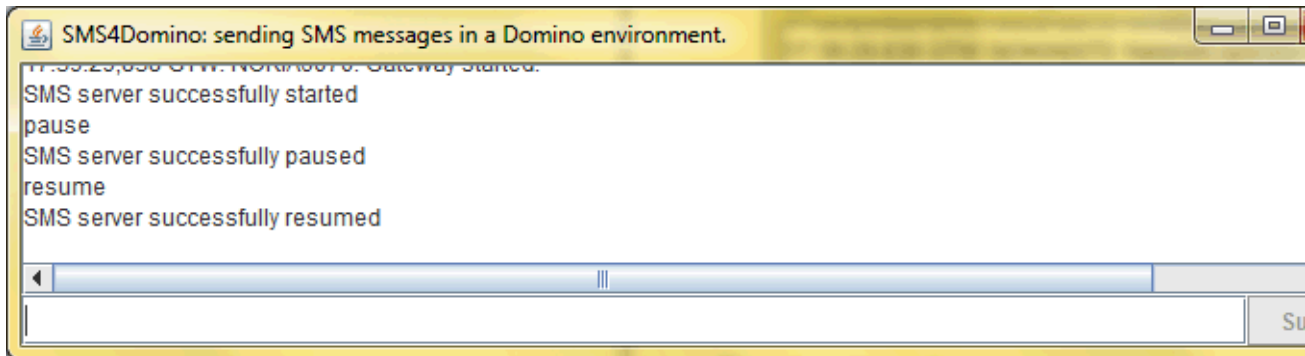
Quit command

Syntax: quit
Purpose: terminates Sms4Domino.
It is recommended to issue the quit command before quitting the Domino server
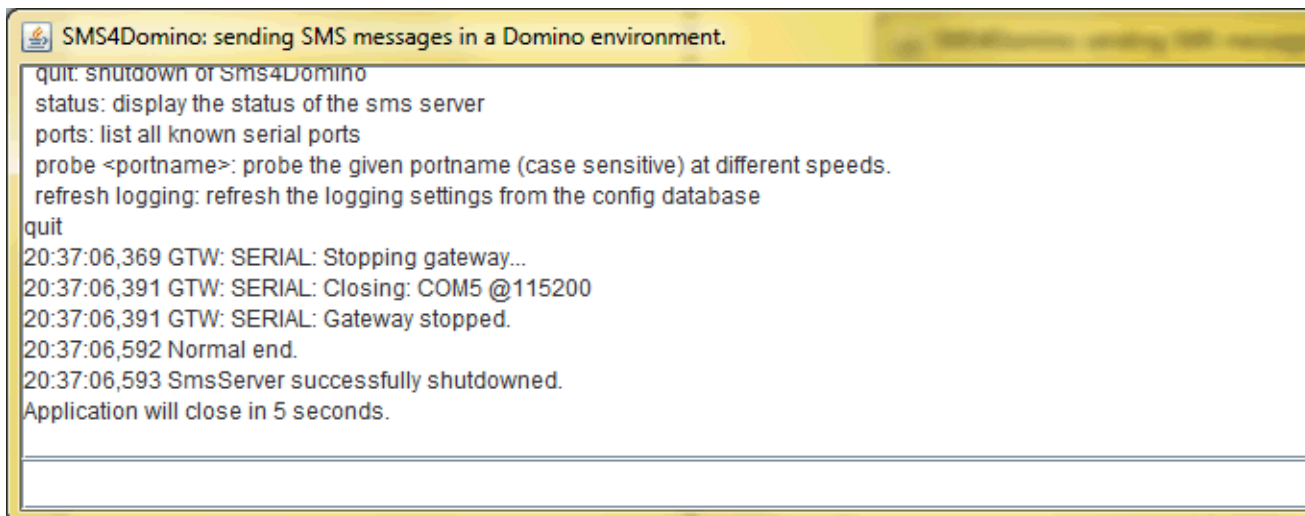
Refresh logging command

Syntax: refresh logging
Purpose: when a Sms4Domino setting is changed (gateway, interface, application profile, license) it is necessary to quit and launch Sms4Domino. The exception is the configuration of the logging. After making a change this command may be used.

```
SMS4Domino: sending SMS messages in a Domino environment.
11-09 14:48:40,476 Notes.ini setting for SMS4DominoConfigDB. Sms4Domino\sms4domino.nsf
11-09 14:48:40,476 No valid notes.ini setting for SMS4DominoInstanceID; default will be used: Server0/Sms4Domino
11-09 14:48:40,480 SMSLib: A Java API library for sending and receiving SMS via a GSM modem or other supported gateway
This software is distributed under the terms of the Apache v2.0 License.
Web Site: http://smslib.org
11-09 14:48:40,481 Version: 3.5.0
11-09 14:48:40,481 JRE Version: 1.6.0
11-09 14:48:40,481 JRE Impl Version: 2.4
11-09 14:48:40,481 O/S: Windows 7 / x86 / 6.1 build 7601 Service Pack 1
11-09 14:48:40,627 Sms4Domino started as stand alone program
11-09 14:48:40,627 SMS4DOMINO - version SMS4DOMINO: version 3.0.1 - released on 09-sep-2012.
help
List of available commands
 help: display this text
 start: read the settings and start the server
 pause: pause inbound and outbound polling
 resume: resume a paused server
 quit: shutdown of Sms4Domino
 status: display the status of the sms server
 refresh logging: refresh the logging settings from the config database
refresh logging
Logging settings have been refreshed.
```

Accounting records

Each time a Sms message was successfully sent or received an accounting document is written. The document contains information about the end user, the date/time of the request, the number of characters, the estimated number of required Sms messages, the gateway and the interface.
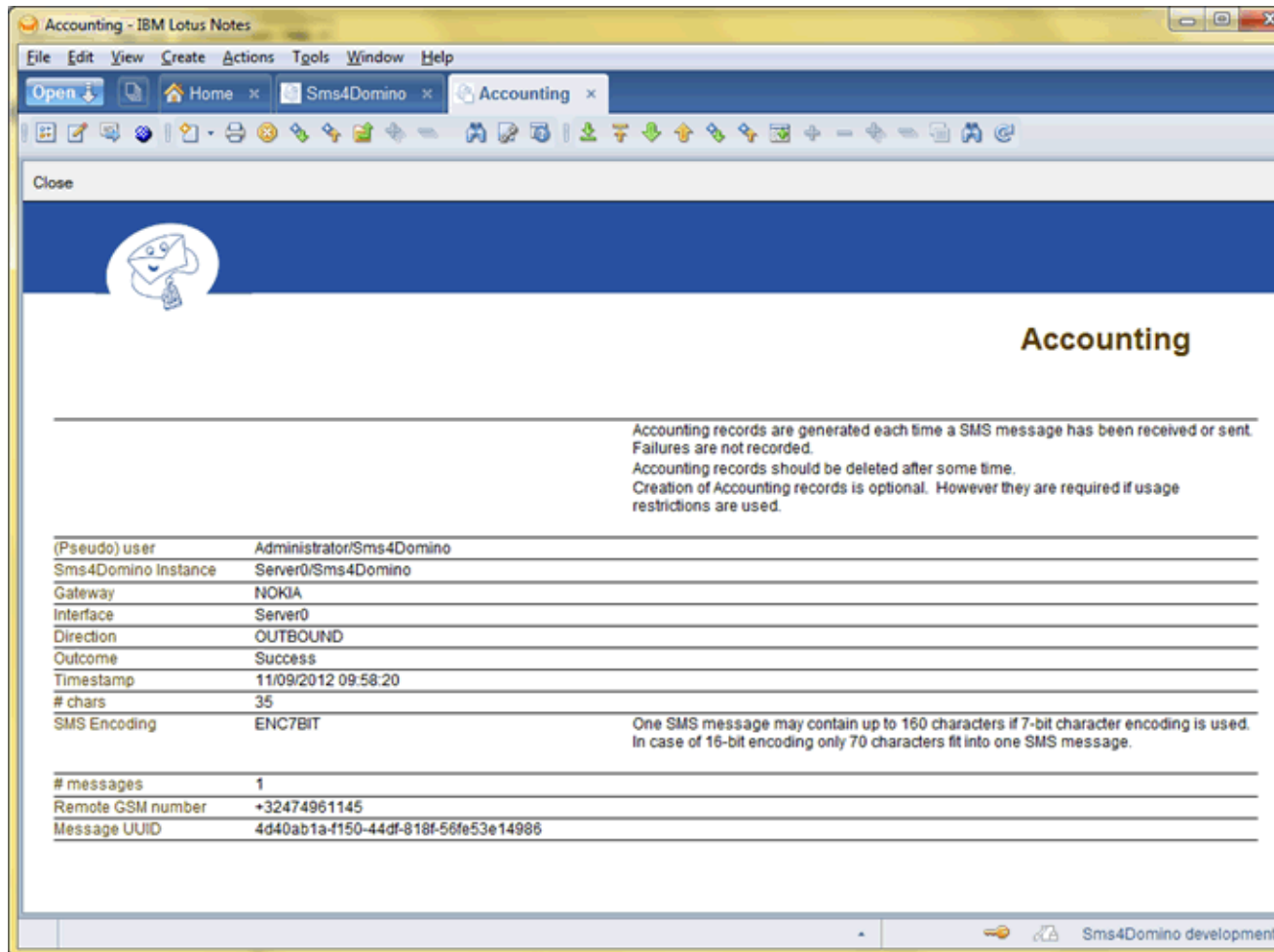
The creation of accounting records may be enabled/disabled in the *Instance Configuration* document.

In parallel with the creation of detailed accounting records the system keeps also aggregated accounting records: one Notes document per gateway, interface and end-user. These documents contain the total volume per day for the corresponsing user, gateway or interface. These aggregated document are only created for the outbound Sms messages.
Every Sms4Domino instance has its own set of aggregated accounting documents.
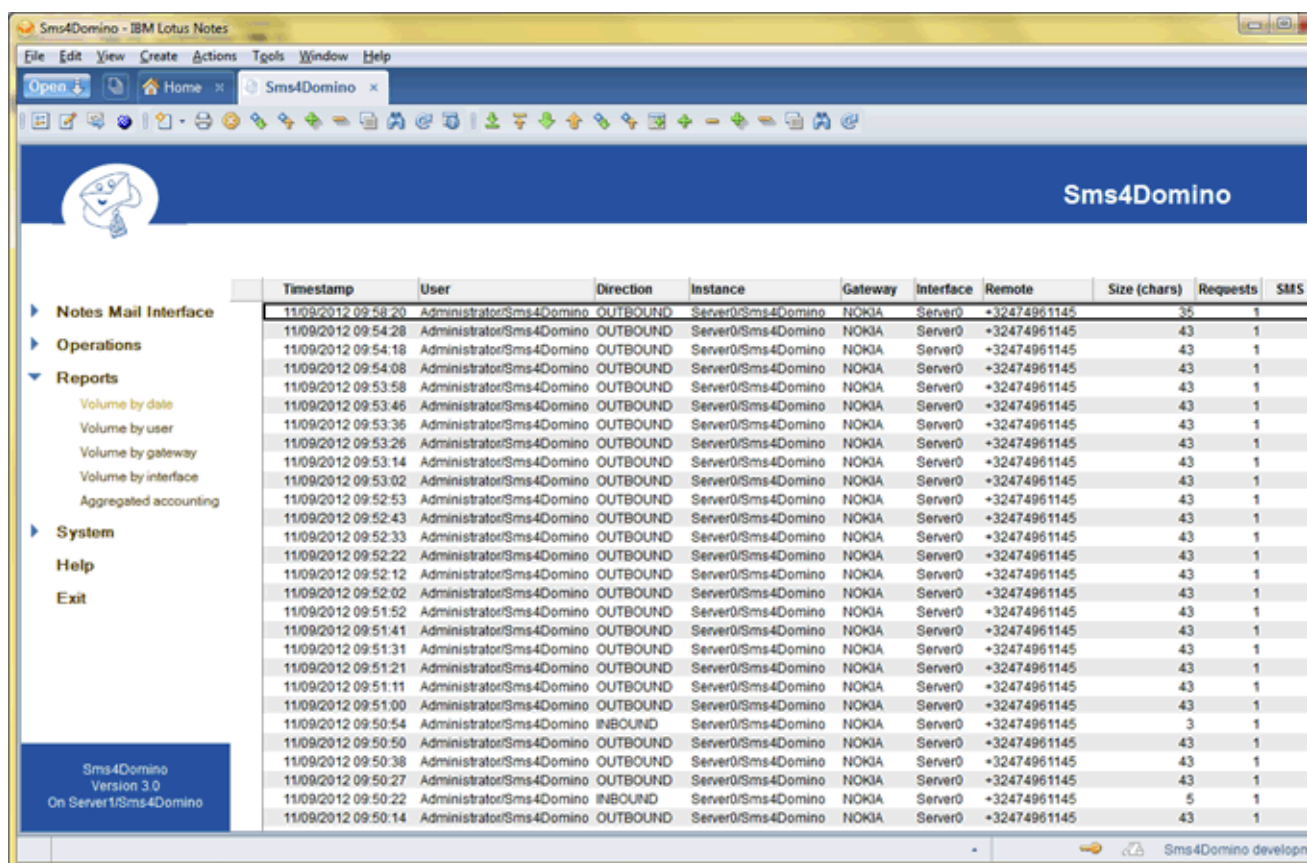
Aggregated accounting documents are required in case restrictions on the volume must be imposed (planned for a future release).

Tip: the administrator should delete the detailed accounting documents when they consume too much disc space.

Volume by date report

This view shows all available accounting records sorted by date .

Every interface may have its own definition of user. For the mail interface the user is the name in the from field of the mail.

Volume by user report

This view shows all available accounting records grouped by user and month.

Every interface may have its own definition of user. For the mail interface the user is the name in the from field of the mail.

Volume by gateway report

This view shows all available accounting records grouped by gateway and month .

Volume by interface report

This view shows all available accounting records grouped by interface and month .

Aggregated accounting report

This view shows all available aggregated accounting documents . For every Sms4Domino instance there may be one document for per gateway, per interface and per user

Health Monitoring

Every running Sms4Domino instance updates a health document at regular times (defined in the Instance Configuration document). This health document is used by the Instance Router when determining the Sms4Domino instance that should send the Sms.

All Health documents are available in the view *System -> Running Instances*.

In above print screen taken at 16:02 we see that two instances are up and running (Server0 and Server1). However the health record for Server2 seems to be outdated. This may be caused by the Sms4Domino instance not running or by a replication failure.

Health documents are created/deleted/updated automatically. There is no user interface for updating them.

Double clicking on one of the documents provide more detail

Sms4Domino Instance

General settings

Sms4Domino supports multi server configurations. Every server may hold at most one instance of the configuration database.
Sms4Domino programs may run on the server (at most one per server) or on a workstation (at most one per workstation). Every running program is identified by an instance id.
Sms4Domino instance documents are created dynamically by Sms4Domino and are the base for routing and failover. They also help in detecting duplicate instance ids.

Instance Id	Server0/Sms4Domino	Unique id of the Sms4Domino instance. The id is supplied as a notes.ini parameter (Sms4DominoInstanceId) when starting Sms4Domino as a standalone program. When Sms4Domino runs as a server add-in program or when the parameter has not been supplied the abbreviated username of the server or client is used.
Session id	ADMR-8Y2H2G	When Sms4Domino starts it generates a random session id. This allows Sms4Domino to detect multiple Sms4Domino instances using the same instance id.
Configuration database	Server0/Sms4Domino!!Sms4Domino\Sms4Domino.nsf	Server!!filename of the configuration database used by this instance. For information only.
Username	Administrator/Sms4Domino	Username that runs this Sms4Domino instance. For information only.
Heartbeat	11/09/2012 16:03:53	A running Sms4Domino server updates this value every x seconds. This allows other instances to determine the availability of the instance.
Active queue	0	Count of current outbound Sms requests (at time of heartbeat)
Total queue	0	Total number of outbound Sms requests (delayed requests included)

- Instance Id: there should be exactly one document for every running instance
- Session Id: When Sms4Domino starts it generates a random session id. This allows Sms4Domino to detect the invalid configuration where multiple Sms4Domino instances using the same instance id.
- Configuration database: Server!!filename of the configuration database used by this instance. For information only.
- Username: Username that runs this Sms4Domino instance. For information only.
- Heartbeat: date/time when the document was updated.
- Active queue: Count of current outbound Sms requests (at time of heartbeat).
- Total queue: Total number of outbound Sms requests (delayed requests included).

Purpose

The Notes Mail Interface is the main path for end users to send SMS messages. Using their standard Lotus Notes® mail client (or any other client that is capable of sending mails via the Domino server) they send mail to a foreign domain that is linked to Sms4Domino.
The mail is processed by Sms4Domino, converted into a SMS message and transmitted via any of the available gateways.

In the following documents we will explain the options for the end user and provide some insight in the processing by Sms4Domino.

Formatting the recipient 's mail address

The most important thing the end user should know is how to address a mail message in order to deliver it as a SMS message. The procedure is slightly different for mail delivered to the server via the Domino router versus mail delivered via the Domino SMTP listener task.

Addressing format when using the Lotus Notes® client, Traveler®, iNotes®, ...

Lotus Notes® uses the @ character to specify the Domino domain name. The administrator is free to choose the name of the foreign domain that corresponds to Sms4Domino. Let's assume the foreign domain is called Sms. The mail recipient should be addressed as

name@number@foreigndomainname (e.g. John Doe@0474965588@sms)
number@foreigndomainname (e.g. 0474965588@sms)

The first part (*name*) is only used as information to the creator of the mail. It is not used at all by Sms4Domino.

The second part (*number*) is the phone number of the recipient. The number is cleaned, transformed and filtered by Sms4Domino. What remains should be a valid number that is accepted by the GSM network. The pre-processing on GSM numbers is explained in a separate document.

The third and last part (*foreigndomainname*) is the name of the foreign domain, linked to the Sms4Domino configuration database, that was specified by the Domino administrator.

Example: John Doe@+32474856978@Sms

Addressing format when sending mail via some SMTP client

Mail addressing should follow the Internet standard RFC 822. The @ character is used to separate the local part from the Internet domain name. Only one @ character is allowed in the address. Domino uses the % character to insert the Domino domain name into the local part of the address.

Suppose an application uses SMTP to generate mail messages via Lotus Domino®. The Domino SMTP listener accepts mail for the Internet domain name mycompany.com. When the application needs to send a SMS to 32474856978 the recipient's address will be

John.Doe%32474856978%Sms@mycompany.com

The sending server should be allowed to relay to the Sms4Domino foreign domain via the receiving server.

Sending a Flash SMS message

A Flash SMS appears directly on the phone's screen, instead of the 'Inbox'. Its an useful alternative to normal SMS when you want to catch the recipients attention immediately. The recipient may have the option of 'Saving' the Flash SMS to his 'Inbox'.

Before sending the message the user should specify the *Delivery options*. Sms4Domino will send *high importance* mail messages as a Flash SMS.
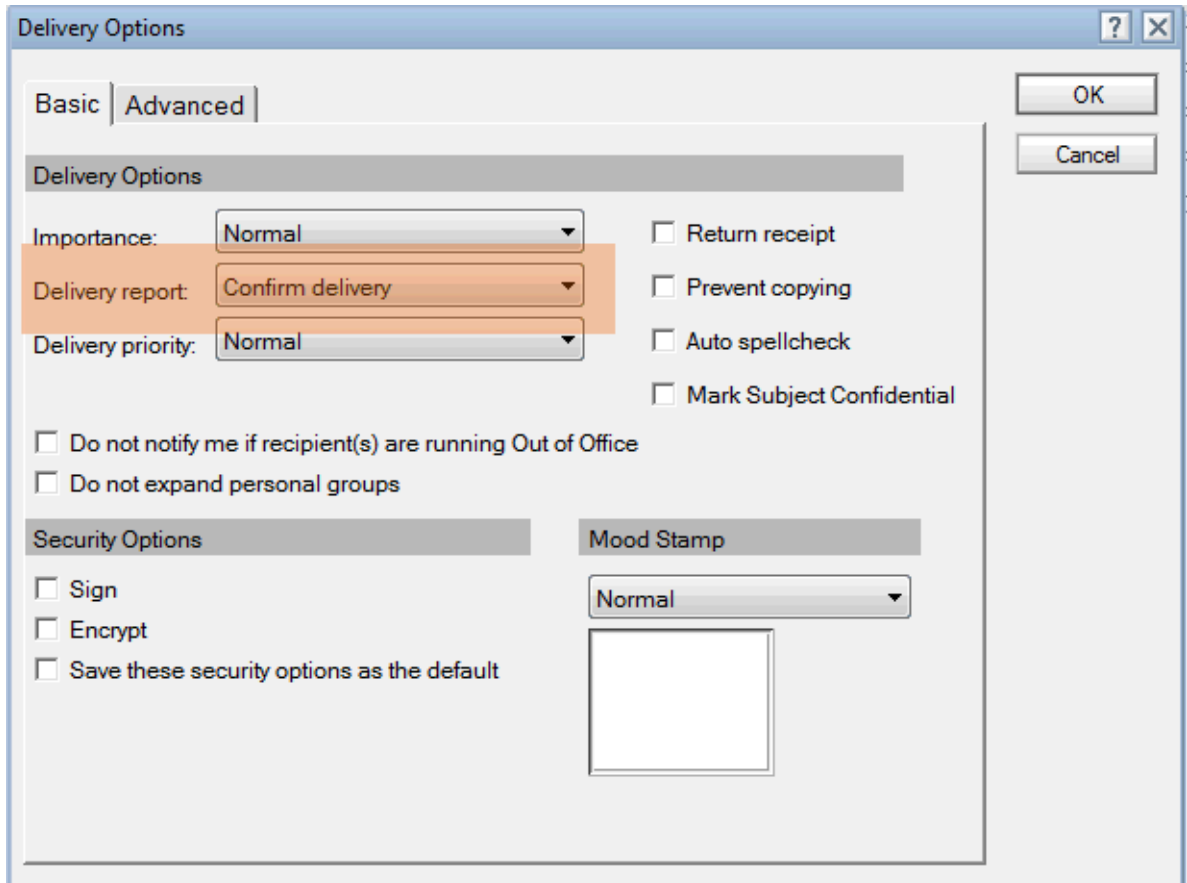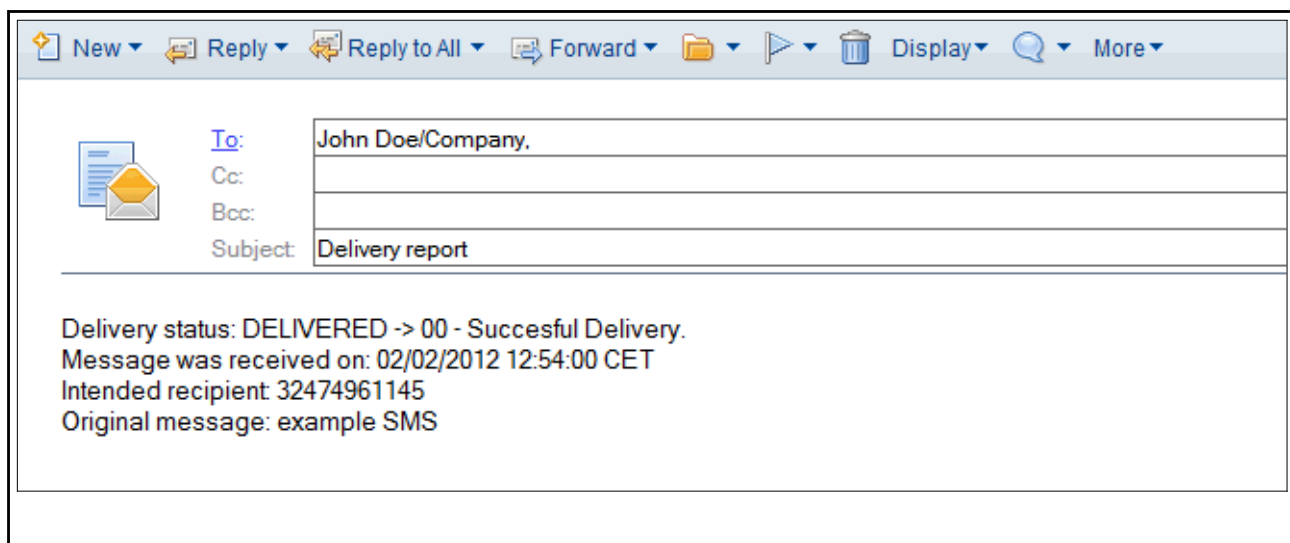
Delivery confirmation

Some GSM networks may return a delivery confirmation when SMS was delivered to the recipient's phone.

The mail setting *Delivery report = Confirm delivery* or *Trace entire path* is recognized by Sms4Domino.

When the operator returns the delivery confirmation or failure to Sms4Domino, this confirmation is converted in a mail message and returned to the sender.

Background information

Delivery reports may not work as expected. All elements involved are described here.

Configuration of the Sms4Domino mail interface document

The mail interface document must have enabled the receipts of status delivery reports

Interface flow	☐ Inbound
	☑ Outbound
	☑ Status delivery report

How is the delivery report matched to the outbound recipient

The end user sends a SMS message to the number 12345678. The GSM network assigns a sequence number to this Sms message. A few minutes later the delivery report is received by Sms4Domino. This delivery report contains the sequence number and the phone number where the Sms was delivered.

The sequence number is only 8 bits wide. So there are only 256 distinct sequence numbers. In heavily used environments Sms4Domino may find more than one outbound Sms with the same sequence number. A better match is achieved by combining the sequence number with the recipient's phone number. However this match may fail because the phone nunber in the delivery report could be different from the phone number in the outbound Sms. E.g. the delivery report may prefix the number with the country code. Sms4Domino requires an exact match between the outbound phone number and the one specified in the delivery report.

In order to be able to match the delivery report with the outbound Sms, Sms4Domino must keep the outbound Sms long enough in the Domino configuration database. This is defined by the setting *Outbound SMS retention time* on the mail interface document.

| Outbound SMS retention time	
Test every n polling intervals	20
No delivery report requested	160 minutes
Delivery report requested	60 minutes

Processing by the mail interface

Mails delivered by the Domino router into the configuration database appear in the view *Notes Mail Interface -> SMS box*
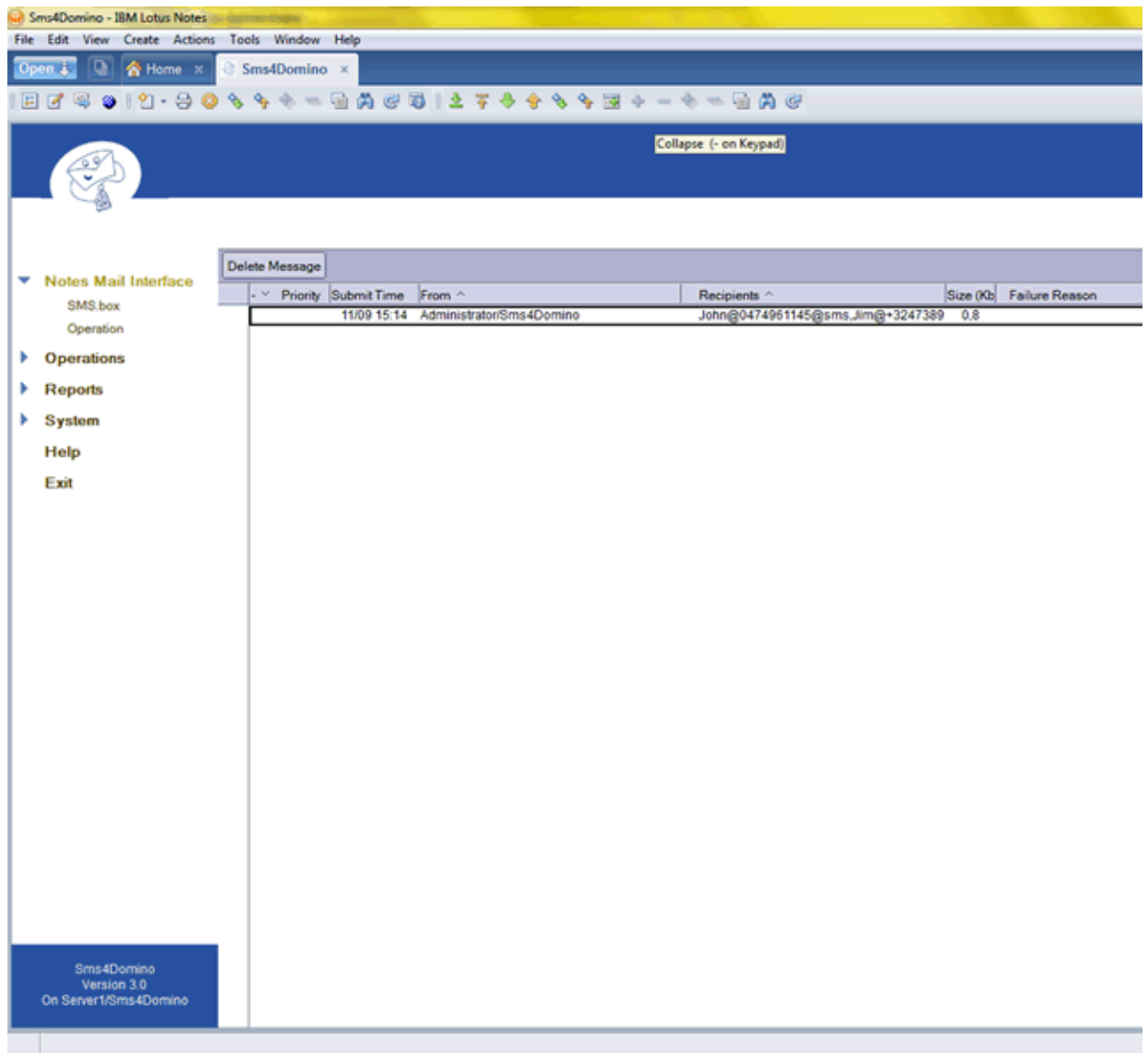
Only the SMS recipients are displayed in the column *Recipients.*

Sms4Domino polls every x seconds the SMS.box. When a new mail is discovered all recipients are extracted. For every recipient a new response document is created. The response documents are visible in the view *Notes Mail Interface -> Operation.*
Based on the *Instance Routing Rules* the Instance responsible for sending the Sms gets assigned.
Sms4Domino may issue a Domino console command to speed up the replication.
In following print-screen both Sms messages will be transmitted by instance Server0/Sms4Domino.
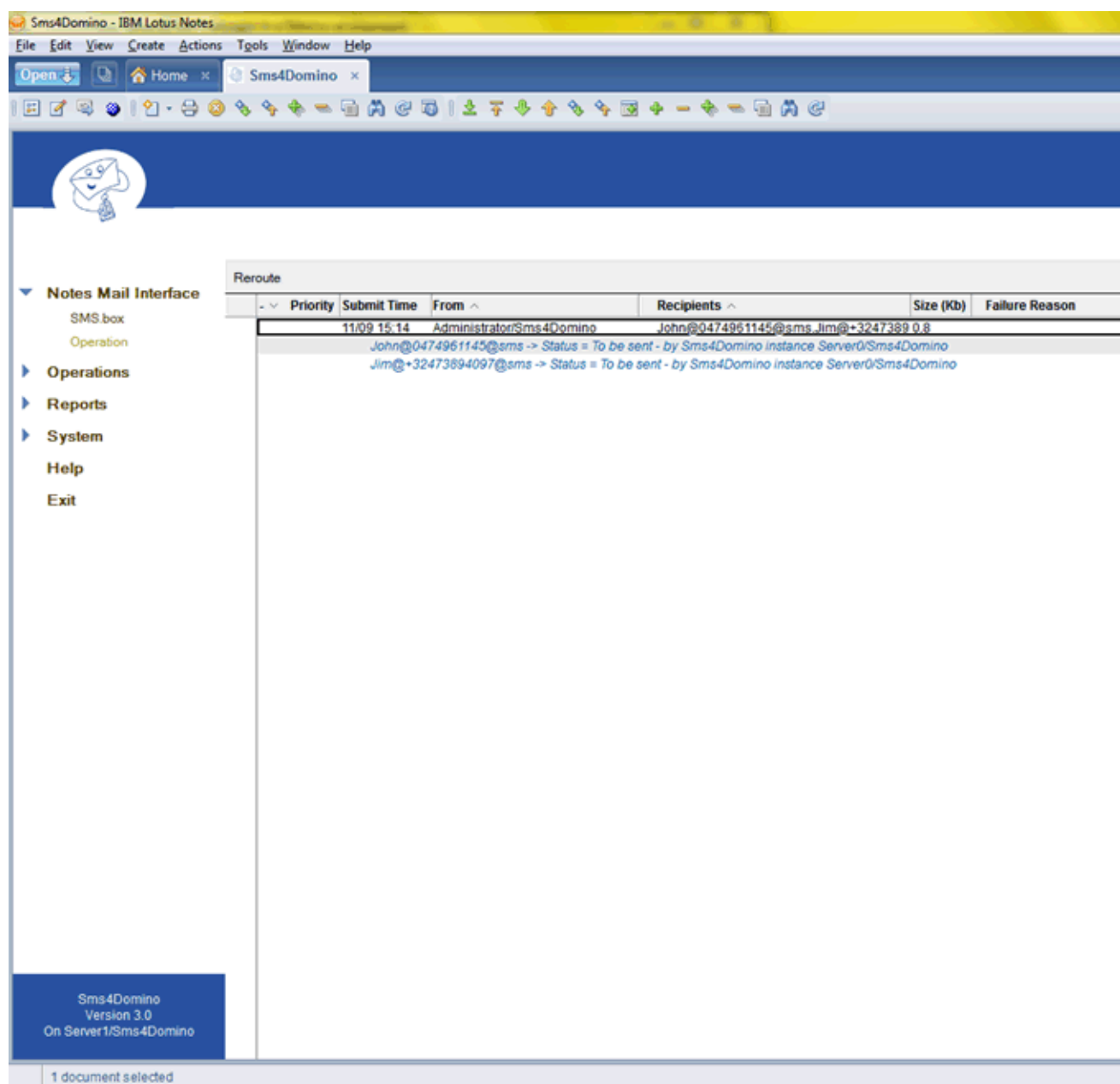
The initial status of the SMS is *To be sent*.

In multi-instance configurations the documents may need to replicate first to another Domino server.
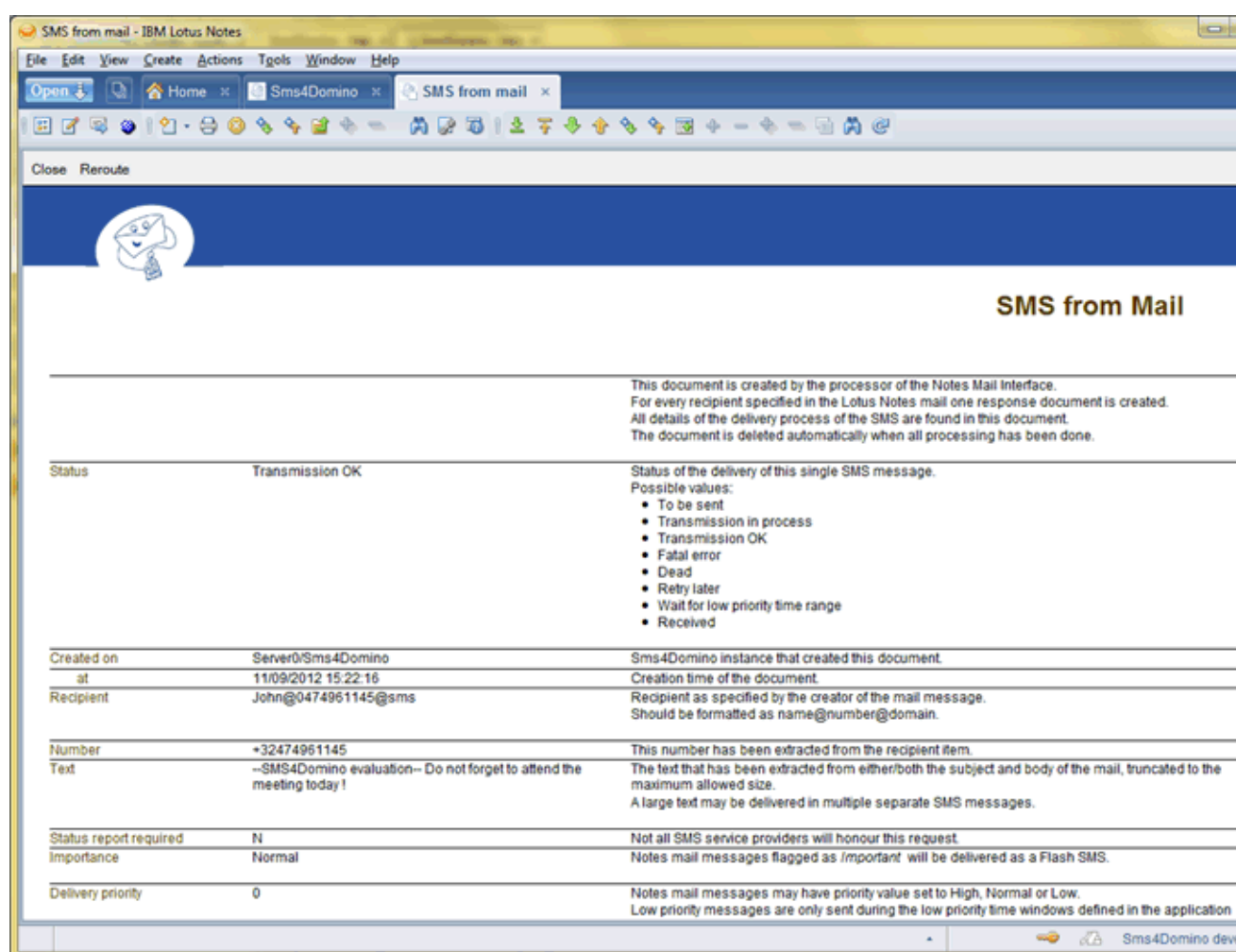
After a few seconds and when a gateway becomes available the transmission is started. The status is then *Transmission in process*.

The Sms response document may be opened (in read mode). All information related to the SMS is displayed and will be updated in real time by Sms4Domino.

After some time the documents in the view are deleted by Sms4Domino.

Reroute Action

Sms4Domino assigns the Instance that is responsible for sending the Sms at the moment the mail is received. In some cases the administrator may want to assign another Instance.

The reroute action could also be used to send a Sms a second time.

- Open the view *Notes Mail Interface -> Operation.*
- Select the documents that must be rerouted.
- Click the action *Reroute.*
- Select one of the defined Sms4Domino instances.
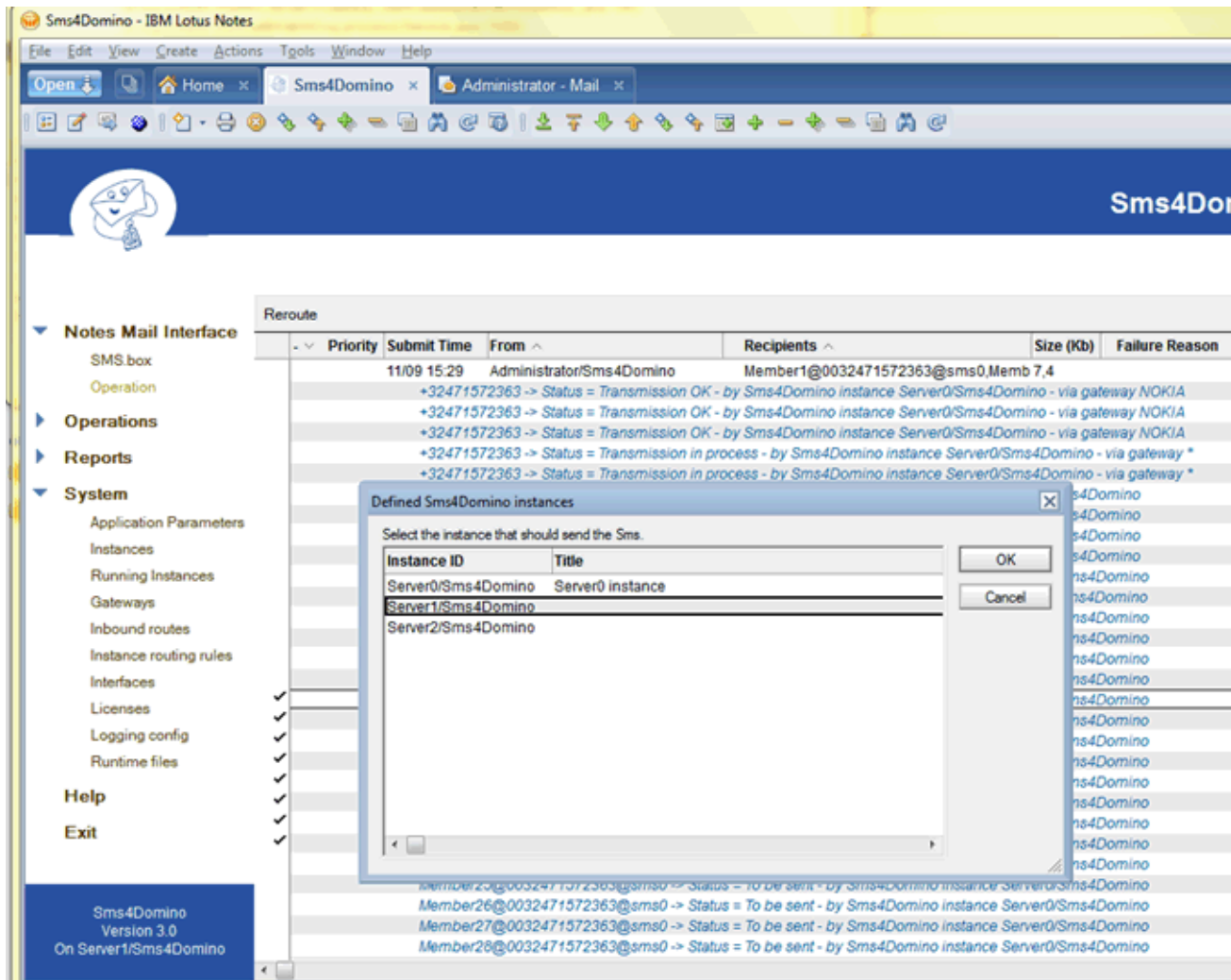
Manually routing received Sms messages

When a Sms message is received from the GSM network it is normally routed based on the available *Inbound Routes*. However, when no valid route is present or when multiple conflicting routes are found the Sms message is kept by Sms4Domino in the configuration database.

All undeliverable inbound Sms messages may be found in the view *Operations -> Received - no match.*

- Select the inbound Sms messages that should be routed.

- Click the action *Forward.*

- Select one or more names from the Domino directory or enter a valid mail address.

- Click *OK.* The Sms message(s) will be delivered by Domino mail.
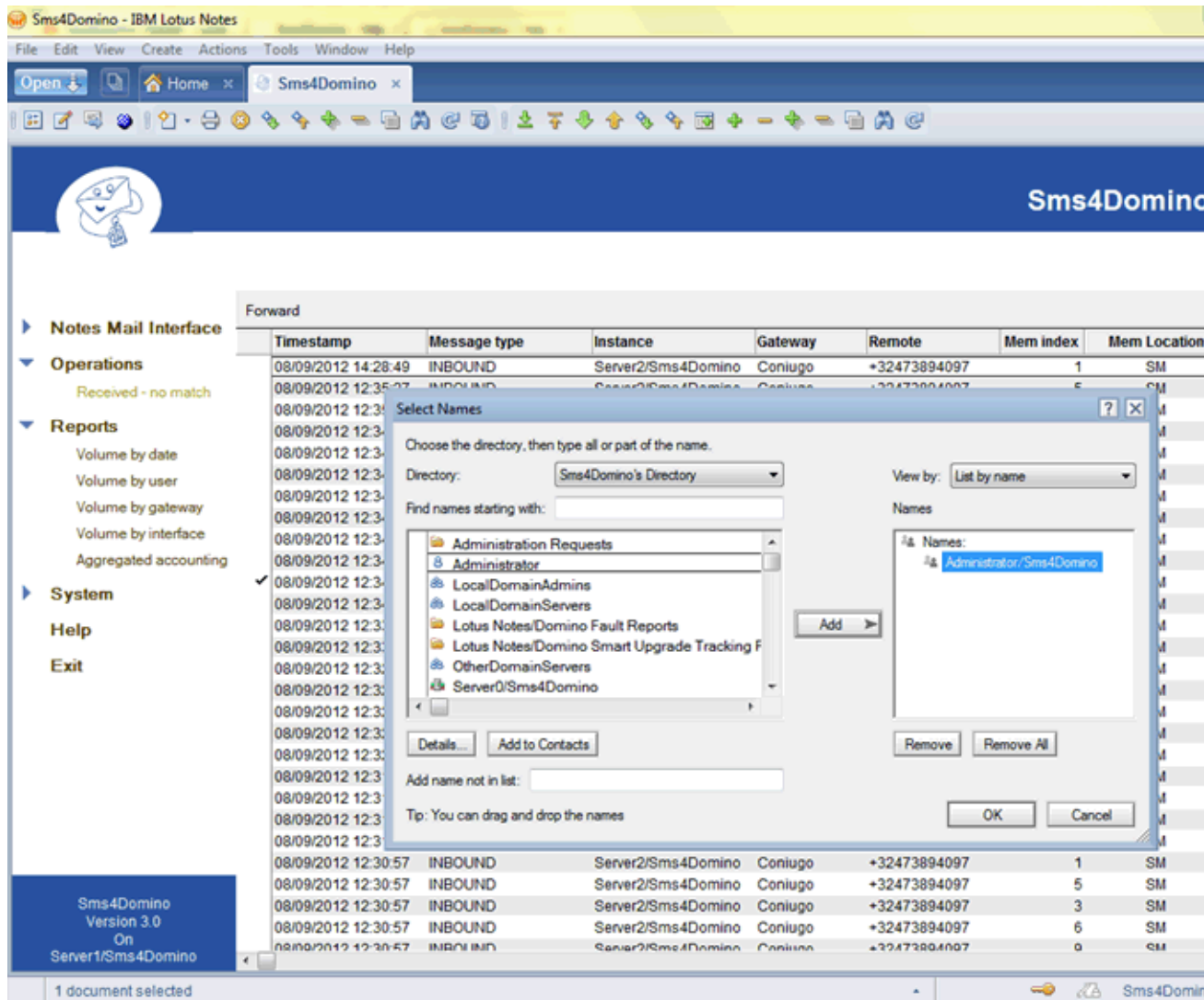
Supported platforms and configurations

Supported operating systems

The current version of Plug-in for Lotus Domino® has been tested on Windows/2003 and Windows/2008 for Intel platforms in 32 and 64 bit mode.

The software requires a Domino R8 server.

For performing administrative actions a Windows based Lotus Note®s clients, version 7 or 8, must be used.

Modems that have been tested with Smslib

Sms4Domino is based on the open source library Smslib. Following modems have been reported to be compatible for use with Smslib. The list of modems recommended and supported by Sms4Domino is more restricted (see Choice and installation of the modem in this help file).

The updated list of compatible modems may be found at http://smslib.org/doc/compatibility/

- Siemens: MC35i, M35, M50, M65, C45, TC35i, C65, M55, TC65t.
- Billionton: PCMCIA PCGPRSQ-B.
- EagleTec: GSM modems.
- Fargo Maestro 20.
- ITengo: 3000, WM1080A.
- Janus: GSM864Q.
- Nokia: 6070, 6100, 6210, 6310, 6310i, 6230, 6230i, 6681, 8250, 8310, 6610, 6800, 7210, 6810, 7250i, 6103, 6020, 3220, 6822, 5140, 5140i, 30 (terminal).
- Multitech: Multimodem GPRS (SMSLib for Java can also drive the Multitech Multimodem via its IP port).
- Sharp: GX30, GX32.
- SIMCOM Ltd: SIMCOM_SIM100S.
- Sony Ericsson: K300i, SE K800i, K700i, K750i, SE W850i, W880i, GC89, Z550a, W800, W580i, W810, i320, GT48, K790i, F305, K770, K618i.
- Ubinetics: GDC201.
- Wavecom: M1206B, M1306B, WMOD2 Wismo, Fastrack Supreme 10, Fastrack Supreme 20, WISMOQCDMA CDMA.
- Huawei: E220 (may require the forced setting of SMSC address), E230.
- Motorola: V3.
- Teltonika: ModemUSB.
- Motorola: V3, L6.
- Samsung: D520, E1410 Guru, GT-I9000.
- Samba: 55-SET GSM/GPRS USB modem.
- Rogers: ZTE 636 USB 'rocket' modem.
- Falcom: USA Samba 75 GSM/GPRS/EDGE (Model MC75).
- BandLuxe: HSDPA c-120.
- SIM548C GSM module.
- Karbonn: K551.
- D-Link: DWM-152.

The following models are reported to be incompatible:

- Nokia: All Symbian models (they partially work or don't work at all).
- Nokia: 6131.
- Blackberry 8700g.
- Nokia 5610 XpressMusic.
- Motorola G18 GSM Modem.

Sms service providers that have been tested with Smslib

Smslib currently supports BulkSMS - all regions - (http://www.bulksms.com) and Clickatell (http://www.clickatell.com). Please note that the above mentioned bulk sms operators are supported only for outbound messaging. SMSLib implements an HTTP/HTTPS interface with them.

How to use Sms 4Domino in IBM® Lotus Notes® Traveler environments

Following info has been copied from IBM's® documentation:

You can set IBM® Lotus Notes® Traveler to use SMS (Short Message Service) mode for Auto Sync, which may greatly increase a device's battery life.

Note: This topic does not apply to Apple devices.
By default, Lotus Notes Traveler maintains a TCP connection between a device and server that allows both to stay in sync with each other. Depending on several factors (such as network provider, NAT timeouts, and device characteristics) maintaining this TCP connection can burden the device's battery, whereas using SMS mode for Auto Sync may decrease the burden on the device's battery and increase the amount of time between battery recharges.

Attention: Do not perform these steps unless you know that you have an unlimited SMS plan; otherwise this may result in a very expensive phone bill.
On the device, go into the Lotus Notes Traveler settings and choose Auto Sync.
Change the Network setting from TCP to SMS e-mail address.
Enter your specific SMS e-mail binding. This is usually your phone number, followed by the @ symbol, and then a carrier-specific domain, for example:
9195555309@messaging.sprintpcs.com or 9195552143@txt.att.net
Click Done.
After following these steps, Lotus Notes Traveler stops maintaining a constant TCP connection to the server, and starts using SMS to notify the device of server-side changes. These SMS messages are received silently by the Lotus Notes Traveler software on the device so they do not interfere with your use of SMS messaging.

Sms4Domino may be used out-of-the-box as the mail-to-sms gateway.

The mail messages generated by Lotus Traveler and addressed to the SMS gateway deviate slightly from standard Lotus Notes mail messages: some common Notes items are missing from the mail message. There is no *form, posteddate* or *subject* item. During the configuration of the filter settings in the mail interface document you should take this into account. Example whitelist filter settings are provided in the interface configuration document.